

IBM Business Process Manager
Version 8 Release 0

*IBM Business Process Manager
Advanced Installation Guide*



PDF books and the information center

PDF books are provided as a convenience for printing and offline reading. For the latest information, see the online information center.

As a set, the PDF books contain the same content as the information center. Some links within the PDF books have been tailored for use in the information centers and may not work properly.

The PDF documentation is available within a quarter after a major release of the information center, such as Version 7.0 or Version 7.5.

The PDF documentation is updated less frequently than the information center, but more frequently than the Redbooks®. In general, PDF books are updated when enough changes are accumulated for the book.

Contents

PDF books and the information center	iii	
Chapter 1. Roadmap: Installing and configuring IBM Business Process Manager Advanced	1	
Chapter 2. Installation types and profiles	5	
Chapter 3. Preparing to install and configure the software	7	
Chapter 4. Planning for IBM Business Process Manager	9	
Planning your path through the installation documentation	9	Topology patterns and supported product features 47
Deciding which configuration to install	9	Determining whether to create a standardized or customized network deployment environment 49
Choosing an operating system	10	Load balancing and failover with IBM HTTP Server 50
Choosing an installation topology	11	Expanding your topology 51
Deciding whether to install into an existing WebSphere Application Server instance	11	Considerations for multiple deployment environments in the same cell 52
Choosing the type of installation	12	Planning your database configuration 53
Deciding what types of profiles to configure	12	Databases and your IBM Business Process Manager topology 54
Choosing whether to install DB2 Express	13	Choosing how and when to configure the common database 55
Choosing whether to use configureNode	13	Supported database types 55
Choosing whether to use PMT or commands to create profiles	13	Database naming restrictions 58
Choosing a method for configuring a network deployment	14	Database time zone and character set considerations 59
Assessing your requirements	14	JDBC drivers and locations 59
Process and process application considerations	15	Non-administrative user considerations 60
Resource considerations	15	Database privileges 61
Development and deployment version levels	16	User ID or schema name privileges 70
Naming considerations for profiles, nodes, servers, hosts, and cells	17	Planning your component-specific database configurations 75
Preparing necessary security authorizations	21	Table and schema creation 77
Installation directories for the product and profiles	21	Planning to configure the common database 78
Choosing the type of installation	24	Planning to configure the Common Event Infrastructure database 83
Installation types and profiles	26	Planning to configure the Business Process Choreographer database 83
Choosing a stand-alone or network deployment environment	27	Planning to configure the messaging engine database 83
Planning your network deployment environment	30	Planning to configure the selector and business rules group database 87
Overview: Deployment environment topologies and patterns	31	Planning to configure Business Process Choreographer 89
Topologies of a network deployment environment	36	Business Process Choreographer overview 89
Single Cluster topology pattern	36	Business Process Choreographer Explorer overview 90
Remote Messaging topology pattern	38	BPEL process archiving overview 90
Remote Messaging and Remote Support topology pattern	40	Shared work items 95
Remote Messaging, Remote Support, and Web topology pattern	43	Planning the topology, setup, and configuration path 96
Customized topology	44	Planning to create a basic sample Business Process Choreographer configuration 101
Considerations for selecting a topology	45	Planning to create a sample Business Process Choreographer configuration including a sample organization 101
		Planning to use the administrative console's deployment environment wizard 102
		Planning for a custom Business Process Choreographer configuration 105
		Planning security, user IDs, and authorizations 106
		Planning the databases for Business Process Choreographer 111

Planning the BPEDB database	112
Planning the messaging engine database	117
Planning the Business Process Archive database	118
Planning for the Business Flow Manager and Human Task Manager	118
Planning for the people directory provider	119
Planning for the Business Process Choreographer Explorer and Business Process Archive Explorer	121
Planning for the Business Process Archive Manager	123
Planning for a remote client application	126

Chapter 5. Installing and configuring IBM Business Process Manager

Advanced 129

Roadmap: Installing and configuring IBM Business Process Manager Advanced	129
Preparing to install and configure the software	132
System requirements	133
Preparing operating systems for product installation	133
Preparing AIX systems for installation	134
Preparing Linux systems for installation	135
Preparing Solaris systems for installation	137
Preparing Windows systems for installation	137
Considerations for HADR setup and configuration	138
Configuring Oracle Data Guard for IBM Business Process Manager	140
Installing and configuring IBM Business Process Manager Advanced	141
Installing and configuring IBM Business Process Manager on Linux.	141
Configuring a stand-alone environment on Linux	142
Typical installation and configuration	142
Custom installation and configuration	166
Configuring a network deployment environment on Linux	553
Granting write permission of files and directories to nonroot users for profile creation or augmentation	553
Creating databases and database design files	555
Configuring the network deployment environment using the configureNode command	602
Configuring profiles and creating a network deployment environment	605
Installing and configuring IBM Business Process Manager on AIX	1002

Configuring a stand-alone environment on AIX	1002
Typical installation and configuration	1002
Custom installation and configuration	1022
Configuring a network deployment environment on AIX.	1401
Granting write permission of files and directories to nonroot users for profile creation or augmentation	1401
Creating databases and database design files	1403
Configuring the network deployment environment using the configureNode command	1450
Configuring profiles and creating a network deployment environment	1453
Installing and configuring IBM Business Process Manager on Solaris	1845
Configuring a stand-alone environment on Solaris	1846
Typical installation and configuration	1846
Custom installation and configuration	1866
Configuring a network deployment environment on Solaris.	2248
Granting write permission of files and directories to nonroot users for profile creation or augmentation	2248
Creating databases and database design files	2250
Configuring the network deployment environment using the configureNode command	2297
Configuring profiles and creating a network deployment environment	2300
Installing and configuring IBM Business Process Manager on Windows	2696
Configuring a stand-alone environment on Windows	2696
Typical installation and configuration	2697
Custom installation and configuration	2722
Configuring a network deployment environment on Windows.	3118
Granting write permission of files and directories to nonadministrative users for profile creation or augmentation.	3118
Creating databases and database design files	3120
Configuring the network deployment environment using the configureNode command	3166
Configuring profiles and creating a network deployment environment	3169

Chapter 1. Roadmap: Installing and configuring IBM Business Process Manager Advanced

The IBM® Business Process Manager Advanced installation and configuration programs accommodate various scenarios. This roadmap guides you through the most common paths for installing and configuring IBM Business Process Manager Advanced in a stand-alone or network deployment environment.

Alternatively, you can use the Interactive Installation and Configuration Guide to answer a series of questions about your target environment. The guide generates a set of installation and configuration topics that are customized to your installation needs.

The roadmap provides the most relevant links for each common installation and configuration scenario, grouped by operating system.

View all | View with tabs

- “Linux”
- “AIX” on page 2
- “Solaris” on page 2
- “Windows” on page 3

Linux

To install IBM Business Process Manager on Linux, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing Linux systems for installation” on page 135

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 142
- “Custom installation and configuration” on page 166
 - “Installing IBM Business Process Manager Advanced interactively” on page 166
 - “Installing IBM Business Process Manager silently” on page 175

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 182

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 605
- Creating a network deployment environment

AIX

To install IBM Business Process Manager on AIX, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing AIX systems for installation” on page 134

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 1002
- “Custom installation and configuration” on page 1022
 - “Installing IBM Business Process Manager Advanced interactively” on page 1022
 - “Installing IBM Business Process Manager silently” on page 1026

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 1032

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 1453
- Creating a network deployment environment

Solaris

To install IBM Business Process Manager on Solaris, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133

- “Preparing Solaris systems for installation” on page 137

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 1846
- “Custom installation and configuration” on page 1866
 - “Installing IBM Business Process Manager Advanced interactively” on page 1866
 - “Installing IBM Business Process Manager silently” on page 1870

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 1876

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 2300
- Creating a network deployment environment

Windows

To install IBM Business Process Manager on Microsoft Windows, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing Windows systems for installation” on page 137

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 2697
- “Custom installation and configuration” on page 2722
 - “Installing IBM Business Process Manager Advanced interactively” on page 2723
 - “Installing IBM Business Process Manager silently” on page 2732

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 2739

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 3169
- Creating a network deployment environment

Chapter 2. Installation types and profiles

There is a relationship between the type of installation you perform, **Typical** or **Custom** and how profiles associated with the installation type are created for each IBM BPM configuration (Express, Standard and Advanced).

The following tables describe the relationships.

- “IBM BPM Advanced”
- “IBM BPM Advanced: Process Server”
- “IBM BPM Standard” on page 6
- “IBM BPM Express” on page 6

IBM BPM Advanced

Table 1. Profile configuration as part of installation: IBM Business Process Manager Advanced

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	Yes

IBM BPM Advanced: Process Server

Table 2. Profile configuration as part of installation: IBM Business Process Manager Advanced: Process Server

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	<i>Process Server only</i>	No	Yes	Not Applicable
Custom	DVD or electronic image	Optionally A feature allows you to create Process Server and or WebSphere ESB profiles for Unit Test Environment (UTE) only	Yes	No	Yes

IBM BPM Standard

Table 3. Profile configuration as part of installation: IBM Business Process Manager Standard

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	No

IBM BPM Express

Table 4. Profile configuration as part of installation: IBM Business Process Manager Express

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	No

Chapter 3. Preparing to install and configure the software

Before preparing to install and configure the software, create a plan for the deployment environment that you want to create.

Use the information listed in the following table to prepare for installing and configuring IBM Business Process Manager.

Table 5. Preparing for installation and configuration

Task	Where to find information	Result after completing the task
Review hardware and software requirements	Depending on your IBM BPM configuration, visit: <ul style="list-style-type: none">• IBM Business Process Manager Advanced system requirements• IBM Business Process Manager Standard system requirements• IBM Business Process Manager Express system requirements	You understand the system requirements necessary to support your IBM Business Process Manager installation.
Prepare your operating system	AIX Preparing AIX systems for installation Linux Preparing Linux systems for installation Solaris Preparing Solaris systems for installation Windows Preparing Windows systems for installation	You have prepared the operating system of each workstation to be used.

Table 5. Preparing for installation and configuration (continued)

Task	Where to find information	Result after completing the task
<p>Make sure you have installed your database management system.</p>	<p>Consult your database documentation for information about installing and administering your database management system.</p>	<p>Your database management system is installed.</p> <p>IBM Business Process Manager embeds the DB2 Express database. If you want to use DB2 Express as your database, you can select it as a feature from the installer and it is installed and configured automatically. The user must have administrative privileges (root or Administrator) to install DB2 Express.</p> <p>Note: If you already have a version of DB2 installed and you want to install DB2 Express, you must uninstall DB2 before running the IBM Business Process Manager installer. If the installer detects a version of DB2 installed and you have selected to install DB2 Express from the installer, you will receive a warning message and will not be able to install DB2 Express.</p> <p>Important: Linux If you are installing DB2 Express as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. See Kernel parameter requirements (Linux) for a list of the kernel requirements. You can locate the current values by parsing the output of the <code>ipcs -l</code> command.</p> <p>Important: You will not be able to properly install and use DB2 Express if the password that is specified does not meet operating system or company restrictions, because an operating system user is created on install.</p>

Chapter 4. Planning for IBM Business Process Manager

To ensure that the system that you implement meets your needs, plan your IBM Business Process Manager before you introduce its software into your enterprise information system.

Planning your path through the installation documentation

The installation process for IBM Business Process Manager, with its various components and potential configurations, supports many scenarios and topologies, ranging from a basic proof-of concept, demonstration, or test environment, to a mature distributed high-availability production environment. You might need some help with the many installation options.

The topics that you need as you work through your installation process are not necessarily together in the navigation tree and, depending on your search query, might not be easy to pick out in the search results.

To address this situation, use the Interactive Installation and Configuration Guide to generate a set of installation and configuration topics that are customized to your precise installation needs. In the Interactive Installation and Configuration Guide form, select the options you need for your installation scenario. As you select each option, the tool automatically removes options that are ruled out by your previous selections. For example, if you indicate that you plan to install the Express configuration, it removes Network Deployment as a potential topology.

After you fill out the form, the tool generates a single topic containing all of the installation and configuration instructions that apply to your scenario. You can save and print these instructions so that they are sharable and portable. You can also go back, change your selections, and generate a new set of instructions. Each time you need to install product components with other options, return to the form and generate a new guide, or use the tool to see what the installation and configuration process looks like when you choose different options.

This following topics describe some of the options presented in the Interactive Installation and Configuration Guide form, the factors you should consider as you plan your installation process, and what causes some options not to be available. The **Help me decide** links in the form bring you to these topics, to help you understand the decisions you make and their impact on the resulting generated instructions.

Deciding which configuration to install

IBM BPM is available in four configurations: Advanced, Advanced: Process Server, Standard, and Express, each targeting various enterprise needs. To select which configuration to install, consider your enterprise's needs in terms of the level of business process management adoption and the product features and topology support that you want.

Table 6. Deciding which IBM Business Process Manager configuration to install

How far along the path to adopting full business process management capabilities is your enterprise?	What specific capabilities and availability support do you require?	Install this configuration:
<p>Project-level adoption: You are starting your first business process management project and are looking primarily for rapid time-to-value and improving user productivity.</p>	<ul style="list-style-type: none"> • Execution compatible with WebSphere Lombardi Edition • IBM Process Designer authoring tool • Collaborative editing and immediate playback • Interactive user interfaces • ILOG-based process rules • Real-time monitoring and reporting • Performance analytics and optimizer • Performance Data Warehouse • IBM Process Center, with a shared asset repository • Up to three process authors and 200 users • Two cores for developers, four cores for the product, but no clustering • Windows and Linux on Intel support 	<p>IBM Business Process Manager Express®</p>
<p>Multiple projects: Your enterprise is involved in multiple business process management projects and requires basic integration support.</p>	<p>All the capabilities of IBM BPM Express, plus the following capabilities and support:</p> <ul style="list-style-type: none"> • Unlimited authors and users • Clustering support for high availability • Linux on System z, AIX, and Solaris support • Network deployment support • DB2® for z/OS® support 	<p>IBM Business Process Manager Standard</p>
<p>Transformative: Your enterprise requires high-volume process automation and needs additional SOA components for extensive service integration and orchestration.</p>	<p>All the capabilities and high-availability support of IBM BPM Standard, plus the following capabilities and support:</p> <ul style="list-style-type: none"> • Execution compatible with WebSphere® Process Server • IBM Integration Designer authoring (BPEL and SOA) • Built-in enterprise service bus (ESB) • Transaction support • Integration adapters • Business Space user interface 	<p>IBM Business Process Manager Advanced</p>
<p>Your enterprise currently uses an earlier version of Process Server alone and you want to update to the latest release; or you want to install IBM Business Process Manager for z/OS.</p>	<p>All the Process Server-related capabilities and high-availability support of IBM BPM Advanced. Does not include Process Designer or Process Center. Available for installing on z/OS.</p>	<p>IBM Business Process Manager Advanced: Process Server</p>

Choosing an operating system

You probably know which supported operating system you plan to install the server components on. Your options might be limited by the product configuration you are installing and your project preferences and available infrastructure. Furthermore, your operating system selection somewhat constrains the options that are available for other choices you make during the installation process.

If you are planning to install IBM BPM Express, you can install the server components on Windows or Linux on Intel systems, or, if you are an IBM Master Data Management customer, AIX. For IBM BPM Standard and IBM BPM Advanced, Windows, Linux on Intel and on System z, AIX, Solaris, and z/OS are supported.

Depending on your choice of operating system, the Interactive Installation and Configuration Guide automatically constrains the options that are available for supported databases, based on probable scenarios. You can choose from the remaining options to generate a custom set of instructions that you can follow when you install and configure the product.

If you choose to install server components on Linux on System z, AIX, Solaris, or z/OS, the generated installation guide includes the instructions for configuring IBM BPM to use an existing database server. If you choose Windows or Linux on Intel, you can choose to use an existing database server, or to have the installation process install and configure DB2 Express, which is useful for proof-of-concept or simple environment installations.

Choosing an installation topology

You can choose to install IBM BPM components all on one computer, which is usually sufficient for small projects, trials, or proof-of-concept installations, or distributed across multiple computers, which is suitable for larger projects and enterprise systems.

You can install a stand-alone topology with any of the product configurations. If you want to install a network deployment topology, you must be installing IBM BPM Standard or IBM BPM Advanced; only stand-alone is available for IBM BPM Express.

If you indicate on the Interactive Installation and Configuration Guide form that you want to install a stand-alone topology, the form greys out options for configuring a network deployment topology, such as using the **configureNode** command or the Deployment Environment wizard, and the generated instructions do not include information about configuring a network deployment environment topology.

If you indicate on the form that you want to install a network deployment topology, you cannot select a typical installation type; installing a distributed topology requires that you choose one of the custom installation types.

Deciding whether to install into an existing WebSphere Application Server instance

You probably know whether you have a WebSphere Application Server instance, possibly with existing profiles that you want to augment, or whether you want to install a new instance.

If you are installing a new test environment or doing a proof-of-concept project, you can install an instance of WebSphere Application Server as part of the larger installation process. However, if you already have a supported version of WebSphere Application Server installed, you can reuse it, and either create new profiles or augment existing profiles, thereby somewhat shortening and simplifying the installation and configuration process. Also, IBM BPM supports only augmentation of deployment manager and stand-alone profiles, or managed node profiles that have not yet been federated to a deployment manager.

If you choose to install into an existing WebSphere Application Server instance, the Interactive Installation and Configuration Guide restricts your options to generating information about the custom installation process in a graphical interface. The options to perform a typical installation or a silent installation with command line or response file are not available because IBM BPM does not document these options for WebSphere Application Server installation. Consequently, if you want to generate a custom installation guide for a typical or silent installation, you must choose to install a new WebSphere Application Server instance.

If you install a new instance of WebSphere Application Server as a nonadministrative or nonroot user, that installation is available to only that user and not other users on the system. See the WebSphere Application Server information center for additional limitations for nonadministrative users.

Choosing the type of installation

The installation type selection determines how you interact with the installation process to provide information and select the options you want.

If you want to install into an existing WebSphere Application Server instance, your only available option is to perform a custom installation through the graphical interface. Generated instructions for performing typical and silent WebSphere Application Server installations are not available in the Interactive Installation and Configuration Guide. If you want to perform a typical installation or a silent installation, you must install and configure a new WebSphere Application Server instance.

A typical installation is the simplest and quickest method for installing and configuring. The installation program uses default values for as many installation options as possible, and you supply values for the remaining options using a graphical interface. This is suitable for a relatively quick proof-of-concept or stand-alone installation and when you are not using an existing WebSphere Application Server instance. The typical installation installs the software and configures a stand-alone profile for you. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Windows The typical installation also installs Process Designer on your machine.

With a custom installation using a graphical interface, you can customize all installation settings.

With a custom installation using a command line, you can run a command that specifies installation settings. This is more suitable if you only require a single installation.

With a custom installation using a response file, you create a file that specifies installation settings, and run an installation command that calls it. Using a response file is a better choice if you know that you need similar installations on several systems in your enterprise.

If you are installing IBM Business Process Manager for z/OS, then there is only one type of installation available, and it is performed using commands.

For all of the custom options, the generated guide includes information about creating profiles after installation.

Deciding what types of profiles to configure

The installation process installs the Process Center and Process Server server components. Process Center provides a repository for process assets, a runtime environment for testing and studying the performance of processes, and a console for administering access to assets and deploying processes to test, stage, or production environments. Process Server is a runtime environment for process applications and a data warehouse for collecting performance data from the applications. It includes administrator consoles for managing and maintaining the runtime environments and data warehouses. To use these server components, you must configure server profiles for them.

You can configure server profiles for either one or both of these, depending on your needs. If you configure one, you can return to these instructions later and configure the other.

If you are installing IBM BPM Advanced: Process Server, the Process Center option is not available.

Configuring profiles for both Process Server and Process Center is two separate activities: you first configure a profile for one, and then another profile for the other. If you want to configure profiles for both Process Center and Process Server, you must select a custom installation type. If you are generating

an installation guide using the Interactive Installation and Configuration Guide form, and you select **Process Center and Process Server**, the generated instructions include information about sequentially configuring both types of profiles.

If you select the typical installation type, you can select only one type of profile to configure. If you want to configure the other profile type as well, you must perform its configuration steps after the typical installation has completed.

Choosing whether to install DB2 Express

You can choose to have the installation process include automatically installing an instance of IBM DB2 Express. Choose this option if you do not have an existing or planned supported database server you intend to use to store content from Process Center or Process Server. If you are installing a test or proof-of-concept environment, you would likely choose to have a new DB2 Express installed automatically, unless one already exists on the system.

Choices you make for other installation options affect what is available for this choice. If you want the generated instructions to include automatically installing DB2 Express, you must install on Windows or Linux on an Intel system. If you are installing any product configuration onto Linux on System z, AIX, Solaris, or z/OS, the installation process does not offer to install DB2 Express; you must use a database server product that you have already installed or will install.

You cannot install DB2 Express on any system that already has DB2 installed on it. Furthermore, you must install as an administrative or root user to install DB2 Express.

The configuration steps configure Process Center or Process Server to use the database of your choice, regardless of whether you automatically install a new DB2 Express instance or you are using an otherwise-installed supported database server.

Choosing whether to use `configureNode`

The `configureNode` command lets you create and extend profiles and typical network deployment environments in a single command line step, as an alternative to the two-step process of using Profile Management Tool or `manageProfiles` to create profiles and the Deployment Environment wizard, the Administrative console, or `wsadmin` commands to configure the network deployment environment.

The `configureNode` command is not available if:

- you select DB2 for z/OS as your database
- you are installing IBM Business Process Manager for z/OS
- you choose to set up a stand-alone environment, for which you do not need to set up a network deployment environment.

If you select **Yes, I want to use the `configureNode` command**, the generated instructions include information about running `configureNode` commands. If you select **No, I will create profiles and the network deployment environment separately**, the generated instructions include information about using one of the two options for creating profiles, and about using one of the three options available for creating the network deployment environment.

Choosing whether to use PMT or commands to create profiles

When creating or augmenting server profiles for Process Center or Process Server, you can choose between using the graphical interface of the Profile Management Tool or the command line interface of the `manageProfiles` command (`zpmt.sh` on z/OS), which facilitates silent configuration.

If you select **Profile Management Tool (PMT)** or **Profiles Management Tool for z/OS (zPMT)**, the generated instructions include information about using the graphical interface tool to create or augment

profiles. If you select **manageProfiles command (silent)** or **zpmt.sh command**, the generated instructions include information about using the command line interface for creating and augmenting profiles.

These options are not available if you choose to perform a typical installation, in which case a profile is created automatically, or if you select to use the **configureNode** command to create the profiles.

Restriction: PMT is not available for installations on Solaris with SPARC 64-bit. The Interactive Installation and Configuration Guide does not enforce this restriction, so if you are installing on that Solaris with SPARC 64-bit, select the **manageProfiles** option.

Choosing a method for configuring a network deployment

You have three options for configuring a network deployment environment, if you are not using the **configureNode** command. You can use the Deployment Environment wizard, the Administrative Console, or the **wsadmin** command. They each configure a deployment environment after you have used the Profile Management Tool or the **manageProfiles** or **zpmt.sh** command to create server profiles.

If you select **Deployment Environment wizard**, the generated instructions include information about using the graphical interface tool to create the network deployment manager. If you select **Administrative Console**, the generated instructions include information about using the WebSphere Application Server Administrative console to create the deployment environment. If you select **wsadmin commands (silent)**, the generated instructions include information about using command line interface to create the deployment environment, facilitating silent configuration.

These options are not available if you choose to perform a typical installation, in which case a profile is created automatically, or if you are installing a stand-alone environment, in which case you do not need a network deployment environment, or if you choose to use the **configureNode** command to create the profiles.

Assessing your requirements

To minimize rework and outages, take the time to study your current environment before you make installation and configuration decisions. Consider your current business requirements and design, the hardware and software already installed, and your current strengths and shortcomings. This planning could also help you minimize your financial investment.

Several factors determine your software needs. These factors can be organized into the following categories.

- Product hardware and software requirements, your own system resource constraints, and the availability of resources to administer and maintain your system
- Applications to be deployed to the runtime environment, and the intended use of the configured environment
- Products, and the versions of these products, to install to meet your requirements

To make wise choices for all these factors, you must understand the following concepts:

- The terminology as it applies to environment configuration
- The administrative architecture of the product that you will install, configure, administer, and maintain
- The available configuration options (through supplied patterns) and how to determine if a pattern addresses your intended use of the product
- The supported methods of implementation, including an understanding of the different task flows for installing the product and configuring the environment

You can use the information in this section to assess and analyze your current and future requirements to develop an environment to meet those requirements.

Important: For the latest information about platform-specific disk space requirements, supported operating systems, and supported database versions, click one of the following links. You can also find operating system fixes and patches that you must install to have a compliant operating system.

- IBM Business Process Manager Advanced system requirements
- IBM Business Process Manager Standard system requirements
- IBM Business Process Manager Express system requirements
- IBM Business Process Manager Tools and Add-Ons requirements

Process and process application considerations

Your current requirements provide the baseline on which to formulate a plan for streamlining integration of your business components. Your vision for the future of your business can provide a guideline that can help you make decisions as your business grows.

You need to know how your product or service is created and delivered. IBM Business Process Manager comes with deployment environment patterns designed to meet the requirements of both production and test environments.

Consider the following:

- Consider how process applications interact with existing services and back-end systems.
- Consider how process applications handle data and how data flows through your system to address a specific business need.

Understand how data persists across retrievals, sessions, processes, and other boundaries when you are developing a solution and configuring its environment.

Consider the following items regarding the process applications to be deployed to your environment:

- Process application invocation patterns

You must understand how the runtime environment handles asynchronous invocations and how the SCA runtime environment leverages the underlying message system to implement asynchronous invocations.

Different applications have different needs. Those needs are determined by factors such as export types, component types, interactions between components, import types, resources needed such as databases or JMS resources, the need for business events, and their transmission mechanism.

- Types of business processes that you plan to implement (transactional business processes, interruptible business processes, non-interruptible business processes)

Non-interruptible business processes, or microflows, are short-running business processes that run in one transaction or without a transaction. Non-interruptible business processes are fast with little effect on performance. All activities within one are processed in a single thread.

Interruptible business processes, or macroflows, are long-running business processes that contain a set of activities, each of which is performed in its own transaction. Interruptible business processes can include activities that require human intervention or calls to remote systems or both.

Asynchronous activities cause a business process to be interruptible because these activities might take minutes, hours, or even days to complete.

Resource considerations

Identify your assets to make the best use of your software and hardware resources and to make informed implementation decisions. Assess your current enterprise information system to determine whether you require any additional hardware or software to meet your business needs.

Consider the following factors:

- Familiarize yourself with current hardware and software. Prepare a list of the available assets.
- Determine the number of physical computer systems that you will use and itemize each piece of physical hardware. Record the following information:

- Amount of installed memory
- Number and type of installed microprocessors
- External media
- Whether a particular unit can be upgraded
- Ensure all servers involved, both for the product and the databases, use the same network time protocol and are set to the same time.
- Itemize the currently installed software and database applications. Record the following information:
 - Function
 - Breadth of use across the company
 - Security requirements
- Prepare a list of your current IT personnel. Determine whether you have the required expertise to install and maintain IBM Business Process Manager, as well as the required expertise to manage your databases. Make sure that the appropriate users have user IDs with the authorizations to successfully install all products and files.

Development and deployment version levels

When you try to determine the version levels of IBM Business Process Manager that you need in your environment, your decision depends on the version levels that were used when your applications were developed. Generally, applications deployed in a previous version of IBM Business Process Manager can run on the next available version of IBM Business Process Manager.

The following table describes compatibility between IBM Business Process Manager V8.0.1, including IBM Integration Designer V8.0.1 (previously WebSphere Integration Developer) and IBM Process Designer V8.0.1, and prior releases.

Task	Supported?
Deployment from WebSphere Integration Developer version 6.1.0, 6.1.2, or 6.2.0 or 7.0.0 or IBM Integration Designer 7.5 to IBM Business Process Manager V8.0.1.	<p>Yes.</p> <p>Important: For WebSphere Adapters V6.1.0, V6.1.2 and V6.2.0, you must install the interim fix titled <i>Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0</i>. If you do not plan to update the WebSphere Adapter to a V7.0 level, and you plan to continue to use the application with WebSphere Adapter V6.1.0, V6.1.2 or V6.2.0, you must apply this interim fix on the source environment.</p> <p>Important: Websphere Adapter for SAP V6.0.2, V6.1.0, V6.1.2 and V6.2.0 are not supported on IBM Business Process Manager V8.0.1. You must update Websphere Adapter for SAP to V7.0 before you can deploy any applications that use Websphere Adapter for SAP on IBM Business Process Manager V8.0.1. For more information specific to WebSphere Adapter for SAP, see Postmigration tasks for IBM Business Process Manager.</p>

Task	Supported?
Running IBM Business Process Manager V8.0.1 artifacts on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5.	<p>No.</p> <p>Applications authored with IBM Integration Designer V8.0.1 cannot be published to or installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5 (any prior release) servers.</p> <p>Applications authored with WebSphere Integration Developer 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Integration Designer 7.5 and then generated in IBM Integration Designer V8.0.1 cannot be published to or installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5 servers.</p> <p>Applications generated using serviceDeploy from IBM Business Process Manager V8.0.1 servers cannot be installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5 servers.</p>

Naming considerations for profiles, nodes, servers, hosts, and cells

This topic discusses reserved terms and issues you must consider when naming your profile, node, server, host, and cell (if applicable). This topic applies to distributed platforms.

Profile naming considerations

The profile name can be any unique name with the following restrictions. Do not use any of the following characters when naming your profile:

- Spaces
- Special characters that are not allowed within the name of a directory on your operating system, such as *, &, or ?.
- Slashes (/) or back slashes (\)

Double-byte characters are allowed.

Windows **Directory path considerations:** The installation directory path must be less than or equal to 60 characters. The number of characters in the *profiles_directory_path\profile_name* directory must be less than or equal to 80 characters.

Note: Use a short path naming convention when you create a profile in a Windows environment to avoid the Windows 255 character path length limitation.

Node, server, host, and cell naming considerations

Reserved names: Avoid using reserved names as field values. The use of reserved names can cause unpredictable results. The following words are reserved:

- cells
- nodes
- servers
- clusters
- applications
- deployments

Descriptions of fields on the Node and Hosts Names and Node, Host, and Cell Names pages: Use the appropriate naming guidelines when you create profiles.

- Stand-alone server profiles
- Deployment manager profiles
- Custom profiles

Table 7. Naming guidelines for stand-alone server profiles

Field name	Default value	Constraints	Description
Node name	<p>Linux UNIX</p> <p>Windows <i>shortHostName</i> Node <i>NodeNumber</i> where:</p> <ul style="list-style-type: none"> • <i>shortHost Name</i> is the short host name. • <i>NodeNumber</i> is a sequential number starting at 01. 	Avoid using the reserved names.	Select any name you want. To help organize your installation, use a unique name if you plan to create more than one server on the system.
Server name	<p>Linux UNIX</p> <p>Windows server1</p>	Use a unique name for the server.	The logical name for the server.
Host name	<p>Linux UNIX</p> <p>Windows The long form of the domain name server (DNS) name.</p>	<p>The host name must be addressable through your network.</p> <p>If you are planning to use Business Space, use a fully qualified host name.</p>	Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.
SSL certificate subject common name (CN)	Generated certificates use the host name as the subject common name (CN).	The common name on the certificate must match the host name for SSL verification.	

Table 8. Naming guidelines for deployment manager profiles

Field name	Default value	Constraints	Description
Node name	<p>Linux UNIX</p> <p>Windows <i>shortHostName</i> Cell ManagerNode <i>Number</i> where:</p> <ul style="list-style-type: none"> • <i>shortHost Name</i> is the short host name. • <i>NodeNumber</i> is a sequential number starting at 01. 	Use a unique name for the deployment manager. Avoid using the reserved names.	The name is used for administration within the deployment manager cell.
Host name	<p>Linux UNIX</p> <p>Windows The long form of the domain name server (DNS) name.</p>	<p>The host name must be addressable through your network. Avoid using the reserved names.</p> <p>If you are planning to use Business Space, use a fully qualified host name.</p>	Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.

Table 8. Naming guidelines for deployment manager profiles (continued)

Field name	Default value	Constraints	Description
Cell name	<p>Linux UNIX</p> <p>Windows <i>shortHostName</i> Cell <i>CellNumber</i> where:</p> <ul style="list-style-type: none"> <i>shortHost Name</i> is the short host name. <i>CellNumber</i> is a sequential number starting at 01. 	<p>Use a unique name for the deployment manager cell. A cell name must be unique in any circumstance in which the product is running on the same physical workstation or cluster of workstations, such as a Sysplex. Additionally, a cell name must be unique in any circumstance in which network connectivity between entities is required either between the cells or from a client that must communicate with each of the cells. Cell names also must be unique if their name spaces are going to be federated. Otherwise, you might encounter symptoms such as a <code>javax.naming.NameNotFoundException</code> exception, in which case, you need to create uniquely named cells.</p>	<p>All federated nodes become members of the deployment manager cell, which you name in the Node, Host, and Cell Names page of the Profile Management Tool.</p>
SSL certificate subject common name (CN)	Generated certificates use the host name as the subject common name (CN).	The common name on the certificate must match the host name for SSL verification.	

Table 9. Naming guidelines for custom profiles

Field name	Default value	Constraints	Description
Node name	<p>Linux UNIX</p> <p>Windows <i>shortHostName</i> Node <i>NodeNumber</i> where:</p> <ul style="list-style-type: none"> <i>shortHost Name</i> is the short host name. <i>NodeNumber</i> is a sequential number starting at 01. 	<p>Avoid using the reserved names.</p> <p>Use a unique name within the deployment manager cell.</p>	<p>The name is used for administration within the deployment manager cell to which the custom profile is added. Use a unique name within the deployment manager cell.</p>
Host name	<p>Linux UNIX</p> <p>Windows The long form of the domain name server (DNS) name.</p>	<p>The host name must be addressable through your network.</p> <p>If you are planning to use Business Space, use a fully qualified host name.</p>	<p>Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.</p>
SSL certificate subject common name (CN)	Generated certificates use the host name as the subject common name (CN).	The common name on the certificate must match the host name for SSL verification.	

Host name considerations:

The host name is the network name for the physical workstation on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote nodes use the host name to connect to and to communicate with this node.

IBM Business Process Manager is compliant to both Internet Protocol version 4 (IPv4) and version 6 (IPv6). Wherever you can enter IP addresses in the administrative console, or elsewhere, you can do so in either format. Note that if IPv6 is implemented on your system you must enter the IP address in IPv6 format, and conversely, if IPv6 is not yet available to you, enter IP addresses in IPv4 format. For more information on IPv6 refer to the following description: IPv6.

The following guidelines can help in determining the appropriate host name for your workstation:

- Select a host name that other workstations can reach within your network.
- Do not use the generic identifier, localhost, for this value.
- Do not attempt to install IBM Business Process Manager products on a server with a host name that uses characters from the double-byte character set (DBCS). DBCS characters are not supported when used in the host name.
- Avoid using the underscore (_) character in server names. Internet standards dictate that domain names conform to the host name requirements described in Internet Official Protocol Standards RFC 952 and RFC 1123. Domain names must contain only letters (upper or lower case) and digits. Domain names can also contain dash characters (-) as long as the dashes are not on the ends of the name. Underscore characters (_) are not supported in the host name. If you have installed IBM Business Process Manager on a server with an underscore character in the server name, access the server with its IP address until you rename it.
- If you are using Secure Sockets Layer (SSL) the host name that the server connects to must match the common name (CN) in the SSL certificate.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. Configuration files for servers do not provide domain name resolution for multiple IP addresses on a workstation with a single network address.

The value that you specify for the host name is used as the value of the hostName property in configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as xmachine.manhattan.ibm.com
- The default short DNS host name string, such as xmachine
- Numeric IP address, such as 127.1.255.3

The fully qualified DNS host name has the advantages of being totally unambiguous and flexible. You have the flexibility of changing the actual IP address for the host system without having to change the server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A disadvantage of this format is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run the server even when disconnected from the network. Define the short name to 127.0.0.1 (local loopback) in the hosts file to run disconnected. A disadvantage of the short name format is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A disadvantage of this format is that the numeric IP address is fixed. You must change the setting of the `hostName` property in configuration documents whenever you change the workstation IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another disadvantage of this format is that you cannot use the node if the host is disconnected from the network.

Preparing necessary security authorizations

Depending on your security policy, you might need a user ID and password to complete tasks such as creating files and folders and accessing the database. Prepare secure user IDs to prevent problems when the servers attempt to access protected data.

- Complete the design of your database.
- Determine the authentication system to use, for example, Lightweight Directory Access Protocol (LDAP).
- Determine what controls are in place that affect the authorizations required for your IBM Business Process Manager installation.
- Identify the systems on which you are installing the product.

The security policies for your site enables global security which indicates that you require specific authorities to install software, create databases or tables, or access databases. To successfully install and operate the product you must do these steps.

To prepare security authorizations for an IBM Business Process Manager database, complete the following steps:

- Prepare a list of user IDs and passwords that have authority to install software on the systems. You must run the installation wizards for IBM Business Process Manager user IDs that have the authority to create files and folders.
- Prepare a list of user IDs, passwords, and roles that are needed for daily operations of the system:
 - Administrative console user IDs and roles to limit capabilities. You can have user IDs for configuring, administering, or monitoring roles.
 - User IDs for each system bus to authenticate system communications.
 - Administrative and monitoring user IDs or groups for each Business Process Choreographer container for authentication with Business Flow Manager and Human Task Manager.
 - User IDs or groups for synchronous calls to authenticate with Business Flow Manager and Human Task Manager.
- Prepare a list of user IDs and passwords that the system uses to access the database tables that it uses during operation.
- Optional: Prepare a list of user IDs and passwords that the system uses to create databases or database tables during installation. Your site policies might restrict this authority to the database administrator. In this case, you must provide generated scripts to the administrator to create the databases or database tables.

You can install and operate your servers in a secure environment.

Installation directories for the product and profiles

The installation directories for IBM Business Process Manager are represented by several variables. The meaning of those variables can differ for a number of factors.

Variables used in the documentation

Several variables representing specific default directories are used throughout the documentation. These file paths are default locations. You can install the product and other components and create profiles in any directory for which you have write access. Multiple installations of IBM Business Process Manager products or components require multiple locations.

Here are the main variables used in the documentation:

install_root

Installation location of IBM Business Process Manager. IBM Business Process Manager is always installed in the same location as the WebSphere Application Server Network Deployment installation with which it is associated.

profile_root

Location of an IBM Business Process Manager profile.

How variable meanings can differ

The meaning of variables used to represent installation directories can differ based on whether you are installing the product on a clean workstation or on a workstation that has an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment. The variables can also differ depending on whether you are performing the installation as a root (Administrator on a Windows system) or nonroot user.

Limitations of nonroot installers

Root, Administrator, and nonroot users can install the product. The default directories the installation program provides differ based on whether the user has root (Administrator) privileges. Root and Administrator users can register shared products and install into system-owned directories (globally shared resources that are available to all users), while nonroot users cannot. Nonroot users can install only into directories they own.

Default directories for Typical Installation

The following tables show the default installation locations of the IBM Business Process Manager base installation and its profiles during a typical installation.

Table 10 shows the default installation root directory into which the installation program installs both IBM Business Process Manager and WebSphere Application Server Network Deployment for both root (Administrator) and nonroot users.

Table 10. *install_root* default directory

Default <i>install_root</i> for root or Administrator users	Default <i>install_root</i> for nonroot users
AIX /opt/IBM/BPM/v8.0	AIX <i>user_home</i> /IBM/BPM/v8.0
Linux /opt/ibm/BPM/v8.0	Linux <i>user_home</i> /ibm/BPM/v8.0
Windows C:\IBM\BPM\v8.0	Windows C:\IBM\BPM\v8.0

Table 11 on page 23 shows the default installation directory for a profile named *profile_name* for both root (Administrator) and nonroot users.

Table 11. *profile_root* default directory

Default <i>profile_root</i> for root or Administrator users	Default <i>profile_root</i> for nonroot users
AIX /opt/IBM/BPM/v8.0/profiles/ <i>profile_name</i>	AIX <i>user_home</i> IBM/BPM/v8.0/profiles/ <i>profile_name</i>
Linux /opt/ibm/BPM/v8.0/profiles/ <i>profile_name</i>	Linux <i>user_home</i> /ibm/BPM/v8.0/profiles/ <i>profile_name</i>
Windows C:\IBM\BPM\v8.0\profiles\ <i>profile_name</i>	Windows C:\IBM\BPM\v8.0\profiles\ <i>profile_name</i>

Table 12 shows the DB2 Express installation location. DB2 Express is installed under the same directory as IBM Business Process Manager.

Table 12. DB2 Express default directory

Default DB2 Express binary location	Database Instance location
Linux /opt/ibm/BPM/v8.0/DB2	Linux Database instance is created under the bpminst user. For example: <i>user_home</i> /bpminst
Windows C:\IBM\BPM\v8.0\DB2	Windows The BPMINST database instance is created under the root (\) of the drive where IBM Business Process Manager is installed. For example, if IBM Business Process Manager is installed under C:\IBM\BPM\v8.0\ then you will see C:\BPMINST.

Default directories for Custom Installation or existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment

The following tables show the default installation locations of the product and its profiles. If you choose to install IBM Business Process Manager on top of an existing supported version of WebSphere Application Server or WebSphere Application Server Network Deployment, IBM Business Process Manager is installed into the same location. Table 13 shows the default installation root directory in such a case for both root (Administrator) and nonroot users.

Table 13. shows the default installation root directory into which the installation program installs both IBM Business Process Manager and WebSphere Application Server for both root (Administrator) and nonroot users.

Default <i>install_root</i> for root or Administrator users	Default <i>install_root</i> for nonroot users
AIX /usr/IBM/WebSphere/AppServer	AIX <i>user_home</i> /IBM/WebSphere/AppServer
Linux Solaris /opt/IBM/WebSphere/AppServer	Linux Solaris <i>user_home</i> /IBM/WebSphere/AppServer
Windows C:\Program Files\IBM\WebSphere\AppServer	Windows <i>user_home</i> \IBM\WebSphere\AppServer

Table 14. shows the default installation directory for a profile named *profile_name* for both root (Administrator) and nonroot users.

Default <i>profile_root</i> for root or Administrator users	Default <i>profile_root</i> for nonroot users
AIX /usr/IBM/WebSphere/AppServer/profiles/ <i>profile_name</i>	AIX <i>user_home</i> /IBM/WebSphere/AppServer/profiles/ <i>profile_name</i>
Linux Solaris /opt/IBM/WebSphere/AppServer/profiles/ <i>profile_name</i>	Linux Solaris <i>user_home</i> /IBM/WebSphere/AppServer/profiles/ <i>profile_name</i>

Table 14. shows the default installation directory for a profile named `profile_name` for both root (Administrator) and nonroot users. (continued)

Default <code>profile_root</code> for root or Administrator users	Default <code>profile_root</code> for nonroot users
Windows C:\Program Files\IBM\WebSphere\AppServer\profiles\ <code>profile_name</code>	Windows <code>user_home</code> \IBM\WebSphere\AppServer\profiles\ <code>profile_name</code>

Default installation directories for IBM Installation Manager

Table 15 shows two default directories related to the Installation Manager tool.

The directories under **Installation directory** are the defaults (per operating system) into which Installation Manager is installed.

The directories under **Agent data location directory** are the defaults (per platform) used by Installation Manager for data associated with the application, such as the state and history of operations performed by Installation Manager.

Values are given for both root (Administrator) and nonroot users.

For more information about the Agent data location, see Agent data location in the Installation Manager documentation. For more information on other defaults for Installation Manager, see Installing as an administrator or non-administrator in the Installation Manager documentation.

Table 15. Installation Manager default installation directories

Defaults for root or Administrator users	Defaults for nonroot users
Installation directory:	Installation directory:
Linux /opt/IBM/InstallationManager/eclipse	Linux <code>user_home</code> /IBM/InstallationManager/eclipse
UNIX /opt/IBM/InstallationManager/eclipse	UNIX <code>user_home</code> /IBM/InstallationManager/eclipse
Windows C:\Program Files\IBM\Installation Manager\eclipse	Windows C:\Documents and Settings\ <code>userID</code> \IBM\Installation Manager\eclipse
	Vista Windows 7 C:\ProgramData\IBM\Installation Manager
Agent data location directory:	Agent data location directory:
Linux /var/ibm/InstallationManager	Linux <code>user_home</code> /var/ibm/InstallationManager
UNIX /var/ibm/InstallationManager	UNIX <code>user_home</code> /var/ibm/InstallationManager
Windows C:\Documents and Settings\All Users\Application Data\IBM\Installation Manager	Windows C:\Documents and Settings\ <code>userID</code> \Application Data\IBM\Installation Manager
Vista Windows 7 C:\ProgramData\IBM\Installation Manager	Vista Windows 7 C:\Users\ <code>userID</code> \AppData\Roaming\IBM\Installation Manager

Choosing the type of installation

Choose the typical installation option to install IBM Business Process Manager interactively in environments for development, quality assurance, and non-clustered staging and production. Choose the custom installation option to install silently, if you will be installing on an existing WebSphere Application Server, or if your business requires complex deployment environment and database topologies.

The typical installation option installs the software and configures a stand-alone Process Center or Process Server profile. A typical installation does not require you to create profiles separately using the Profile Management Tool or **manageprofiles** utility.

The custom installation can also be performed interactively, or by choosing a silent installation. Choose an interactive installation if you want to see the prompts as you make your decisions. Choose the silent installation if you want to be able to script the installation in future, or if you want to use the same response file on multiple systems.

After using either the typical or custom installation, you can use the `configureNode` command to configure a deployment manager and one or more custom (managed node) profiles and create a pattern-based network deployment environment.

If you use the custom installation, you must either run `configureNode` or use the Profile Management Tool or **manageprofiles** command-line utility to configure a stand-alone profile, or one or more deployment manager and custom (managed node) profiles. After profile creation, you can use the deployment environment wizard to generate a pattern-based network deployment configuration.

Tip: The Interactive Installation and Configuration Guide is a tool in this information center that you can use to generate a set of installation and configuration topics customized to your precise installation needs. You can save the generated output, print it, take it with you when you install on another computer, or share it with someone else on your team. Each time you need to install with other options, return to the form and generate a new guide, or use it to see what the installation and configuration process looks when you choose different options.

Typical installation

The typical installation option is the simplest and quickest method for installing and configuring the software. It is also the ideal method for getting full Business Process Manager features and functionality (for Express, Standard, or Advanced configurations) on a single machine.

With a typical installation, you can maintain a personal copy of IBM Business Process Manager on your system. You can use it to develop integration services or business processes locally, or you can contribute artifacts using the export and import functions in the product. A typical installation is also recommended for proof of concept work and for learning about the features and functions of the product.

This installation option sets most settings by default, but allows you to specify the installation path and database type.

- A typical installation creates a typical stand-alone profile for IBM Process Center or IBM Process Server.
- A typical installation can install DB2 Express if an existing database is not specified. DB2 Express is available only for Windows and Linux 32 and 64 bit systems, and the user must have administrative privileges (Administrator or root) and must not have another DB2 product installed.
- A typical installation automatically populates the database.
- **Windows** On the Windows operating system, a Process Center installation automatically installs IBM Process Designer.
- A typical installation creates shortcuts for the Process Center Administrative Console, Playback Server Administrative Console, Process Designer (on Windows), and WebSphere administrative console.

Custom installation

Choose custom installation for the following reasons:

- You want to install on an existing installation of WebSphere Application Server.
- Your deployment environment requires additional cluster nodes.
- Your business requires complex deployment environment and database topologies.

- You prefer a silent installation.

Installation types and profiles

There is a relationship between the type of installation you perform, **Typical** or **Custom** and how profiles associated with the installation type are created for each IBM BPM configuration (Express, Standard and Advanced).

The following tables describe the relationships.

- “IBM BPM Advanced” on page 5
- “IBM BPM Advanced: Process Server” on page 5
- “IBM BPM Standard” on page 6
- “IBM BPM Express” on page 6

IBM BPM Advanced

Table 16. Profile configuration as part of installation: IBM Business Process Manager Advanced

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	Yes

IBM BPM Advanced: Process Server

Table 17. Profile configuration as part of installation: IBM Business Process Manager Advanced: Process Server

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	<i>Process Server only</i>	No	Yes	Not Applicable
Custom	DVD or electronic image	Optionally A feature allows you to create Process Server and or WebSphere ESB profiles for Unit Test Environment (UTE) only	Yes	No	Yes

IBM BPM Standard

Table 18. Profile configuration as part of installation: IBM Business Process Manager Standard

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	No

IBM BPM Express

Table 19. Profile configuration as part of installation: IBM Business Process Manager Express

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Typical using Launchpad	Electronic image only	Yes	No	Yes	Not Applicable
Custom	DVD or electronic image	No	Yes	No	No

Choosing a stand-alone or network deployment environment

Choose a stand-alone environment to evaluate the product or to support development of applications and services. Choose a network deployment environment when your production environment needs additional features such as capacity, availability, scalability, and failover support.

A stand-alone environment is the easiest to install and configure, and requires little planning. A network deployment environment needs more extensive installation and configuration tasks that can involve several roles.

For a network deployment environment, you should carefully plan the characteristics with a goal of meeting the requirements of the work that business applications and services are to perform on it. There are multiple aspects to consider, including the following:

- Number of physical workstations and hardware resources that you require
- Number of clusters and cluster members required to support your business
- Number of databases required
- Authentication roles and security considerations
- The method that you will use to implement the deployment environment
- Other supporting resources such as a user registry (for security), one or more HTTP servers (for web content), necessary firewalls, load balancers, and so on.

Stand-alone environment

You can use a stand-alone environment to deploy service component architecture (SCA) modules in one server process.

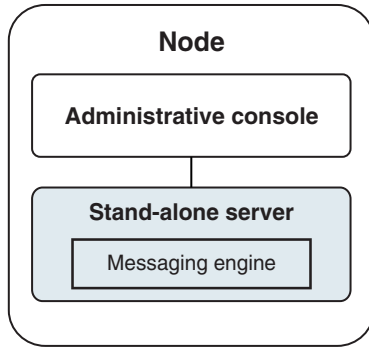


Figure 1. A stand-alone environment

To evaluate the product or to support development of applications and services, you can install samples to deploy a sample solution to the stand-alone server. You can explore the resources used for this sample in the administrative console.

To start with a stand-alone environment and then to include it into a network deployment environment, federate it into a deployment manager cell. You can do so only if no other nodes have been federated to that cell.

When you install the product software, you can choose to create the profile for a stand-alone development environment (**qwps**). The profile that is created is suitable only in a test scenario or to support application development. For a scenario in which you want a stand-alone server environment for production purposes, install the product software. Then use the Profile Management Tool or **manageprofiles** command-line utility to configure the stand-alone profiles.

Network deployment environment

A network deployment environment contains a collection of interconnected servers and clusters to run your service requester and provider enterprise applications and their mediation modules. The environment can also include application servers on WebSphere Application Server.

A network deployment environment provides a collection of interconnected servers and clusters that support application components:

- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Business rules
- Mediations
- Relationships

The environment also supports servers for WebSphere Enterprise Service Bus and WebSphere Application Server.

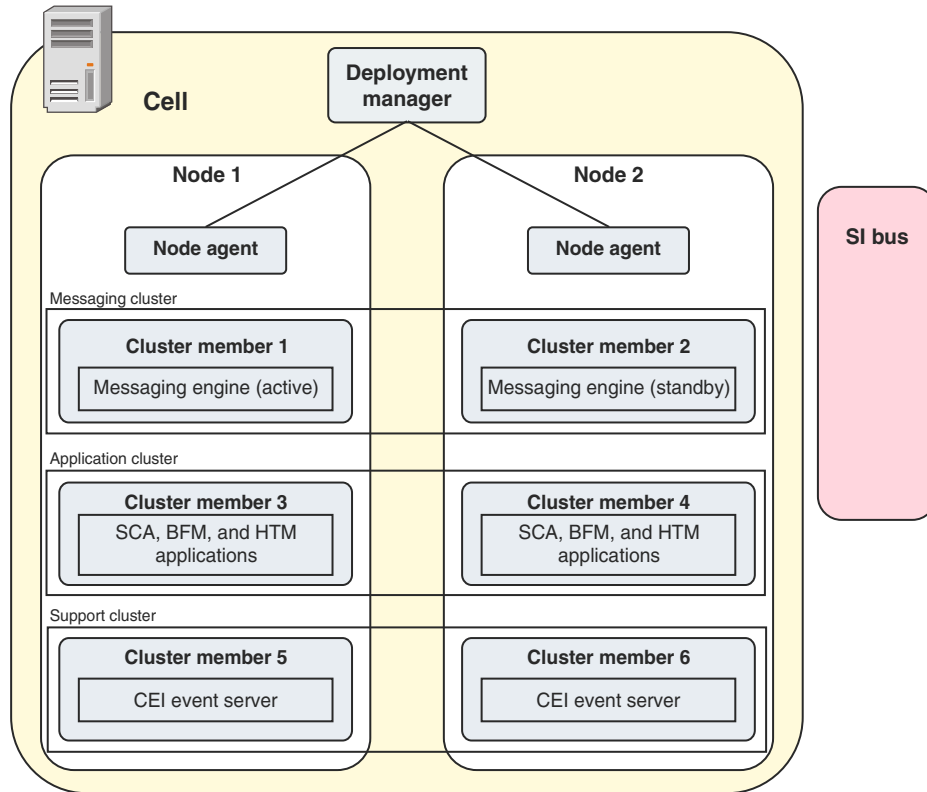


Figure 2. A network deployment environment

The servers and clusters run on one or more managed *nodes*, each of which corresponds to a logical or physical computer system.

Servers can be grouped into *clusters* to support load-balancing and failover.

A deployment environment of interconnected servers or clusters provides performance, availability, scalability, isolation, security, and stability characteristics that cannot be provided by a stand-alone server. In addition, you can manage all the servers or clusters from a centralized *deployment manager*.

A complete collection of servers and clusters managed by a deployment manager is configured and managed as a *deployment environment*.

To install a network deployment environment, install the product software, and then configure profiles for a deployment manager and one or more custom (managed) nodes. Later, you can create the deployment environment to be managed. You can create a *standardized* deployment environment from provided topology patterns, or you can configure clusters and servers to create a *customized* deployment environment.

How intended usage affects your choice of stand-alone or network deployment cluster topology pattern

The following table shows how the intended use of IBM Business Process Manager affects your choice of stand-alone or a network deployment cluster topology pattern, and the associated amount of planning involved:

Table 20. Choice of stand-alone or network deployment cluster topology pattern for intended use of IBM Business Process Manager

Intended use	Configuration path and planning activities
A single server Unit Test Environment (UTE)	The stand-alone profile configuration path, with little planning required.
A clustered test environment	Standard Remote Messaging and Remote Support topology pattern of network deployment environment, with little planning required.
A production environment, with good flexibility	Standard Remote Messaging and Remote Support topology pattern of network deployment environment, with little planning required.
A highly optimized production environment	A customized topology that addresses unique processing requirements and business requirements. Detailed planning required as described in this section of the documentation.

For more information about selecting an appropriate cluster topology pattern, refer to the related concepts links.

Planning your network deployment environment

Setting up a network deployment environment involves many decisions, such as the number of physical workstations and the type of pattern you choose. Each decision affects how you set up your deployment environment.

Before you plan your deployment environment complete the following tasks:

- Choose a database type
- Identify available resources
- Identify necessary security authorizations

When you plan the layout of interconnected servers, you must make some decisions. These decisions influence trade-offs that you make between the available hardware and physical connections, the complexity of the management and configuration and requirements such as performance, availability, scalability, isolation, security, and stability.

1. Identify the functional requirements of the deployment environment.
 - a. Identify the features or runtime capabilities of your deployment environment.
Consider whether the deployment environment will support other IBM Business Process Manager products and non-BPM products in addition to IBM Business Process Manager.
 - b. Identify the component types that you will deploy.
Consider the component types and the interactions between components as part of the requirements.
 - c. Identify the import and export implementation types and transports.
Consider the resources needed for the databases or Java™ Message Service (JMS) resources and the need for business events and their transmission mechanism.
 - d. Identify any functional requirements that are not related to applications.
Consider security servers, routers, and any other hardware or software requirements to handle business events.
2. Identify the capacity and performance requirements for your environment.
3. Decide on the number of physical servers that you need for each function.
4. Design your deployment environment.

Decide on the pattern. For IBM Business Process Manager, you can select one of four established topology patterns:

- Single Cluster
- Remote Messaging
- Remote Messaging and Remote Support
- Remote Messaging, Remote Support, and Web

If none of these patterns meets your needs, you can use the administrative console to create a custom deployment environment.

Note: If your configuration supports multiple IBM Business Process Manager and non-BPM products in addition to, and compatible with, IBM Business Process Manager, the patterns of those products would be available to you when you create your deployment environment.

For more information about the patterns and the differences between them, see “Topologies of a network deployment environment” on page 36.

5. Understand the methods available to you for configuring your deployment environment.

You can configure the following types of deployment environments for IBM Business Process Manager:

- A standardized network deployment environment

A standardized network deployment environment is based on a topology pattern template included with the software and implemented by using the Deployment Environment configuration wizard or `wsadmin` commands.

You can use the Deployment Environment Configuration wizard to create clusters with the Single Cluster, Remote Messaging, Remote Messaging and Remote Support, and (if applicable) Remote Messaging, Remote Support, and Web cluster topology patterns.

- A customized network deployment environment

A customized network deployment environment is a configuration that you create from the administrative console, as opposed to a "template-based" configuration from the Deployment Environment wizard.

You should create a customized network deployment environment only if the topology patterns that are included with the software do not meet your configuration needs.

As is the case with the standardized environment, you can create a customized network deployment environment with `wsadmin`.

Overview: Deployment environment topologies and patterns

A network deployment environment can have many topologies, and can be created from several standard topology patterns.

What is a topology?

A topology is the physical layout of the *deployment environment* required to meet your business needs for capacity, availability, and scalability.

You can set up topologies for both the Process Center and Process Server components of IBM Business Process Manager.

Many factors affect how you design and implement your topology. For example, you must consider business and application requirements, resource requirements and constraints, the intended purpose of the environment, and the operating system.

IBM Business Process Manager includes patterns for the following topologies, which you can use to address many business scenarios, from proof-of-concept (POC) to a fully functional production environment:

- Single Cluster
- Remote Messaging
- Remote Messaging and Remote Support
- Remote Messaging, Remote Support, and Web

Each topology pattern has certain design characteristics that address a particular business need. For example, on distributed systems, the Single Cluster topology pattern is typically used for a testing or proof of concept scenario. On z/OS systems, this topology pattern is the default pattern and can be used in production environments.

The design characteristics of each topology have been captured as *topology patterns* that are supplied as configuration templates with the product.

You are not obligated to use a standardized (IBM-supplied) topology pattern. If none of the topology patterns address your specific need, you can create a custom topology pattern.

The purpose of deployment environment patterns

A deployment environment topology pattern specifies the constraints and requirements of the components and resources involved in a deployment environment. There are IBM-supplied topology pattern for each topology layout. These topology patterns provide rules and guidelines for component interaction that are characteristic of the most commonly used BPM topology patterns. The IBM-supplied topology patterns are based on well-known and tested configuration scenarios. They contain a repeatable and automated method of creating a deployment environment. Each topology pattern is designed to meet the configuration requirements and business needs of the associated topology. Using topology patterns helps you create a deployment environment in the most straightforward way.

Because the deployment environment topology patterns represent recommended topologies with component configurations that work together, you can be sure that you are building a fully functional deployment environment. You can use the configuration rules of a deployment environment topology pattern to generate a fast path configuration. This action is possible because many design decisions are implemented in the topology pattern; for example, which components to configure, and which default parameters and resources are needed.

Each supplied deployment environment topology pattern addresses a specific set of requirements. Most requirement sets can be met when you use one of these topology patterns. To select a topology pattern, complete all of the following steps:

- Understand the requirements of the business solution that you are creating.
- Review and understand the capabilities and characteristics of the IBM-supplied topology patterns.
- Decide which topology pattern to use.

If none of the IBM Business Process Manager topology patterns suit your needs, you can use the administrative console or scripting (wsadmin commands) to create a customized topology pattern.

Databases and deployment environments

Before you create and configure a network deployment environment, you must configure your database and create the required database tables. At a minimum, to use IBM Business Process Manager, you need to configure the following databases on your database management system:

- The Common database (CMNDB)
- The Process Server database (BPMDB)
- The Performance Data Warehouse database (PDWDB)

For a stand-alone server configuration, use the **Typical** installation option to configure these databases and create the required database tables automatically.

For a network deployment environment (customized deployment environment or standardized deployment environment), you or your database administrator must configure the databases outside the installer. Additional databases are required to support additional functionality. For example, if your IBM Business Process Manager configuration includes Business Process Choreographer, Business Space, or Common Base Event monitoring, you or your database administrator must configure these databases and use supplied utilities or scripts to create the required database tables. You must do this configuration before you can create the network deployment environment.

For more information, see Planning your database configuration.

Functions of IBM-supplied deployment environment topology patterns

Any IBM Business Process Manager deployment contains a basic set of functions that together form a complete production environment.

To design a robust deployment environment, you must understand the functionality that each cluster can provide in an IBM-supplied topology pattern or custom deployment environment. You can allocate a specific type of function (for example, the support infrastructure function) to a particular cluster. Understanding the functions can help you choose the deployment environment topology pattern that best meets your needs.

For network deployment, clusters can collaborate to provide specific functionality to the environment. Depending on your requirements, you assign specific functions to each cluster within the deployment environment, to provide performance, failover, and capacity.

The clusters configured in a deployment environment provide the following functions.

The functions can exist in a single cluster, or can be spread across multiple clusters. Each standardized (IBM-supplied) topology pattern creates a different number of clusters to support the functions. The number of clusters in your deployment environment depends on the topology pattern that you are using.

Application deployment target

An application deployment target is the set of servers (cluster) to which you install your applications (for example, human tasks, business processes, and mediations). Depending on which deployment environment topology pattern you choose, the application deployment target might also provide messaging infrastructure and supporting infrastructure functions. Select the appropriate product depending on the type of applications that you intend to deploy.

- If the applications contain human task or business process artifacts, install Process Server.
- If the applications contain mediation modules only, install WebSphere Enterprise Service Bus.

In a Single Cluster topology pattern, the application deployment target provides the entire functionality of the deployment environment.

Supporting infrastructure

The supporting infrastructure includes the Common Event Infrastructure (CEI) server and other infrastructure services used to support your environment and manage your system. These services include:

- Business rules
- Selectors
- Human tasks
- Business processes

Important: You must use a custom profile with the same product functionality for this node as you did for the application deployment target cluster.

The business rules are not tied to the Supporting infrastructure cluster. In fact, business rules can exist and work everywhere in the cell. The business rules administrative function (performed from the Business Rules Manager) can be deployed on the supporting infrastructure cluster (in a three cluster configuration) or in the web application infrastructure cluster (in the four cluster configuration). The same principle applies to the human tasks and business processes. The human tasks and business processes run on the application deployment target cluster, because that is where the human task and business process containers are configured. However, you administer processes and tasks from the Business Process Choreographer Explorer, which can reside on the supporting infrastructure cluster (in a three cluster configuration) or on the web application infrastructure cluster (in a four cluster configuration).

Messaging engine infrastructure

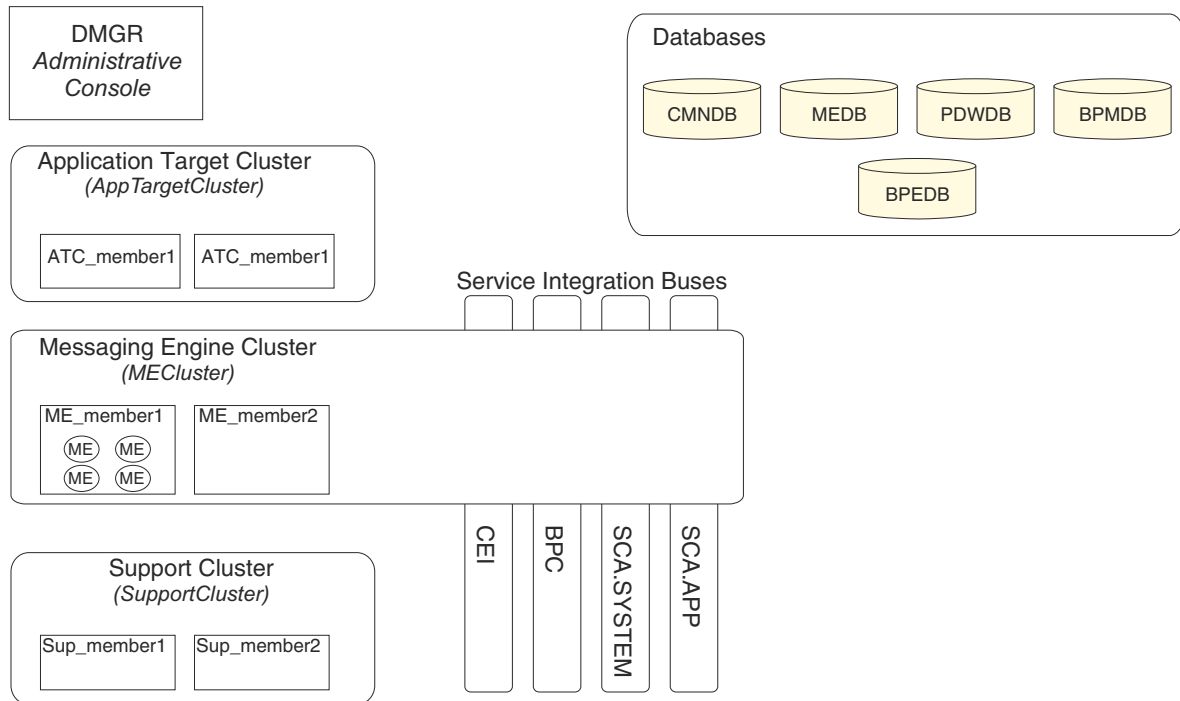
The messaging infrastructure is the set of servers (cluster) where the messaging engines are located. The messaging infrastructure is used to provide asynchronous messaging support for your applications and for the internal messaging needs of the IBM Business Process Manager components. The messaging engines enable communication among the nodes in the deployment environment. Your cluster can consist of members on nodes created with WebSphere Application Server instead of IBM Business Process Manager if the cluster solely provides the messaging function.

Web application infrastructure

Consists of a cluster where the web-based components Business Process Choreographer Explorer, Business Rules Manager, Business Space, Process Portal, and REST API services are located.

For topologies in all environments, the fundamental pieces of IBM Process Server and WebSphere ESB are always similar. In all IBM Process Server and WebSphere ESB cells, the deployment manager is the central point of administration for the cell.

The following diagram illustrates the points of interest in an IBM Process Server deployment environment configured using a Remote Messaging and Remote Support topology pattern. This topology pattern is identical to the Process Center deployment environment, except that the Process Center deployment environment has a Process Center console in the application deployment target cluster. The Process Server network deployment environment does not include a Process Center console.



This diagram includes:

- **Deployment Manager (DMGR):** A server that manages operations for a logical group or cell of other servers.
- **Application Target Cluster:** The deployment target for customer applications. This cluster is configured to provide functionality for BPEL processes, human tasks, process applications, and SCA. In a Process Center deployment, this cluster also includes the Process Center console.
- **Databases:**
 - Common database (CMNDB) - Required. Contains tables shared on a cell-wide basis for multiple IBM BPM Advanced Process Server and WebSphere ESB capabilities.
 - Process Server database (BPMDB) - Required. Contains tables for IBM BPM Advanced Process Server capabilities.
 - Performance Data Warehouse database (PDWDB) - Required. Contains tables for Performance Data Warehouse capabilities.

- Messaging Engine database (MEDB) - Each messaging engine requires a unique set of database tables.
- Business Process database (BPEDB) - Each deployment target that is configured for Business Process Choreographer requires a set of tables for Business Process Choreographer functions.
- **Service Integration Buses (SIBuses):** At least one server or cluster is a member of each of the SIBuses. On multiplatforms, the messaging engine cluster is a member of each of the SIBuses. The bus member hosts a single messaging engine (ME) for each SIBus that it is associated with.
- **Messaging Engine (ME) Cluster:** Hosts the messaging engines. The ME Cluster is a member of each of the four service integration buses.
- **Support Cluster:** The Support Cluster hosts the applications that provide some utility for, but do not contribute to the workload of, the Application Target cluster. For example, the Support Cluster might host the Business Process Rules Manager.

Topologies of a network deployment environment

A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

Related concepts:

“Considerations for selecting a topology” on page 45

Selecting an appropriate topology for your deployment environment depends upon several factors.

“Topology patterns and supported product features” on page 47

A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

Single Cluster topology pattern

The *Single Cluster* topology pattern is an IBM-supplied topology pattern. In a Single Cluster topology pattern, all the functions of the deployment environment are combined into a single cluster.

This is the default pattern for IBM Business Process Manager for z/OS.

A Single Cluster topology pattern is ideal for limited hardware. Because all the components are installed in the same cluster, fewer physical machines are required. However, because each server instance must run the supporting applications and your integration applications, you need more memory for the individual Java Virtual Machines (JVMs). In addition, one or more members of the cluster must also run the messaging engines required for asynchronous interactions. Thus, the Single Cluster topology pattern is typically used for proof of concept, development, and testing environments.

Combining all aspects of the IBM Business Process Manager environment into a single cluster has other implications aside from the increased memory requirements.

- Because asynchronous interactions (involving JMS and MQ/JMS bindings), human tasks, state machines, and long-running business processes can make extensive use of the messaging infrastructure, a single cluster environment is not ideal for applications with these components.
- Any messaging requirements must be kept to a minimum with this topology pattern (except for z/OS).
- Service Component Architecture (SCA) internal asynchronous invocations, the Java Message Service (JMS), and MQ messaging bindings do not support multiple messaging engines in the same cluster.

If necessary, choose one of the other topology patterns in which the messaging infrastructure is in a separate cluster from the application deployment target.

The Single Cluster topology pattern is suitable for scenarios that are focused on running applications and on synchronous invocations. This topology pattern is also not ideal if you intend to make extensive use of the Common Event Infrastructure (CEI). Generating events and CEI-related messaging traffic, places an additional burden on the cluster members.

From an administrative and scalability perspective, the Single Cluster topology pattern has advantages. A single cluster where each member runs all the IBM Business Process Manager components are simpler to administer. Instead of several server instances in multiple clusters, you have a single cluster with fewer members. If the needs of your environment grow, scaling the infrastructure is a simple matter of adding additional nodes and cluster members. Thus, the process of adding capability is simple, but all components are scaled at the same rate. For example, each additional cluster member adds CEI processing whether you need it or not. If the messaging engines spread across server members use policies, there could be some additional administrative effort in creating and maintaining the policies.

In a Single Cluster topology pattern, all deployment environment functions and functional groups of components run on a single cluster:

- The applications:
 - Business Process Choreographer (BPC) container
 - Process Server
 - Process Portal
- The messaging bus members:
 - Business Process Choreographer bus members
 - Common Event Interface (CEI) bus members
 - Performance Data Warehouse (PDW) bus members
 - Process Server bus members
 - Service Component Architecture (SCA) application bus members
 - SCA system bus members
- The support infrastructure applications:
 - CEI server application
 - Performance Data Warehouse
- The web applications:
 - Business Process Choreographer Explorer
 - Business Rules manager
 - Business Space
 - Process Portal
 - REST API Services

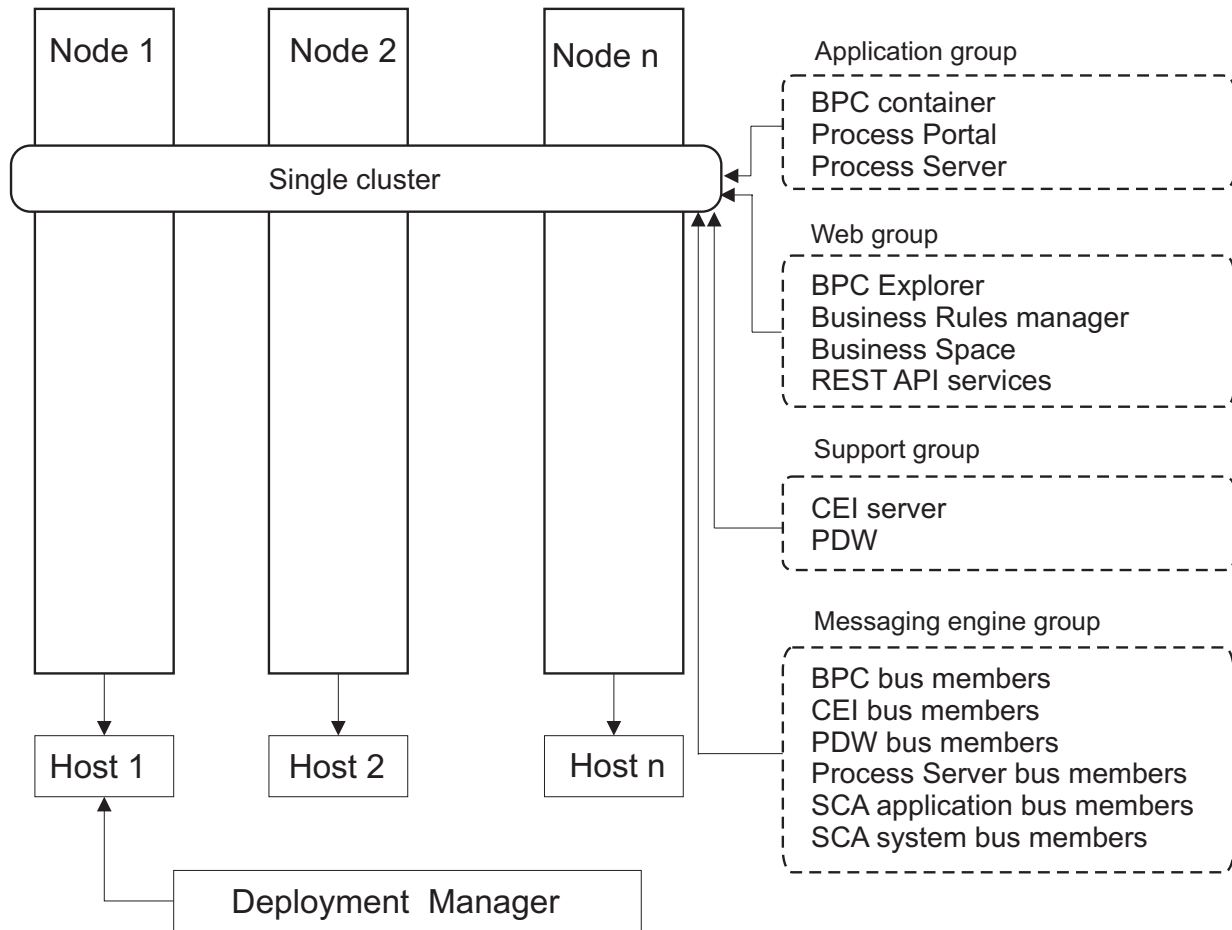


Figure 3. Single cluster topology pattern

Remote Messaging topology pattern

The *Remote Messaging* topology pattern is an IBM-supplied topology pattern. In a Remote Messaging topology pattern, the deployment environment functions are divided between two separate clusters.

The Remote Messaging topology pattern provides a separate cluster for the messaging function. This topology pattern is suitable for scenarios involving asynchronous invocations, because the cluster can be scaled for this load. The components are divided between the two clusters.

For environments that must support numerous human tasks, long-running business processes, state machines, and asynchronous interactions, a Remote Messaging topology pattern has advantages over the Single Cluster topology pattern.

Separating the messaging infrastructure into a separate cluster removes the messaging overhead from the application target cluster. When you have a separate messaging infrastructure, you need less memory for the application target cluster members. This topology pattern also differs from the Single Cluster topology pattern in terms of the hardware required. Because there are two clusters with multiple cluster members, the hardware requirements are greater for distributed environments.

From an administrative perspective, the requirements for the Remote Messaging topology pattern are greater than the requirements for the Single Cluster topology pattern. Additional clusters and additional

cluster members increase the administrative effort required. In addition, because you are distributing the messaging engines across the members of the messaging cluster, you must create and maintain policies.

In the Remote Messaging topology pattern, the supporting applications and the Common Event Interface (CEI) components are still part of the application target cluster. Thus, for environments that make extensive use of CEI, the Remote Messaging topology pattern might not be ideal either. For small to medium-sized businesses, or for businesses without extensive monitoring or auditing requirements, this topology pattern is generally suitable.

The scalability options for the Remote Messaging topology pattern are as straightforward as the options for the Single Cluster topology pattern. Because the messaging engines are subject to one of n policies (each messaging engine is active on only one server), adding additional members to the messaging cluster has little effect. When you use policies to spread the messaging engines across server members, you can divide the messaging burden across a maximum of three servers. (The SCA.SYSTEM and SCA.APPLICATION engines are active on the same server.) Thus, adding more than three cluster members to the messaging cluster has no effect on the processing capability of the messaging infrastructure. Scaling the application target cluster is relatively easy. If you need additional processing capability for your applications or for the supporting infrastructure, you can add additional nodes and members to the application target cluster.

In a two-cluster topology pattern, the messaging members run on the messaging cluster and all other all deployment environment functions and functional groups of components run on a the application deployment target cluster.

The application deployment target cluster hosts the following:

- The applications:
 - Business Process Choreographer (BPC) container
 - Process Server
 - Process Portal
- The support infrastructure applications:
 - CEI server application
 - Performance Data Warehouse
- The web applications:
 - Business Process Choreographer Explorer
 - Business Rules manager
 - Business Space
 - Process Portal
 - REST API Services

The messaging infrastructure cluster hosts the following:

- Business Process Choreographer bus members
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members

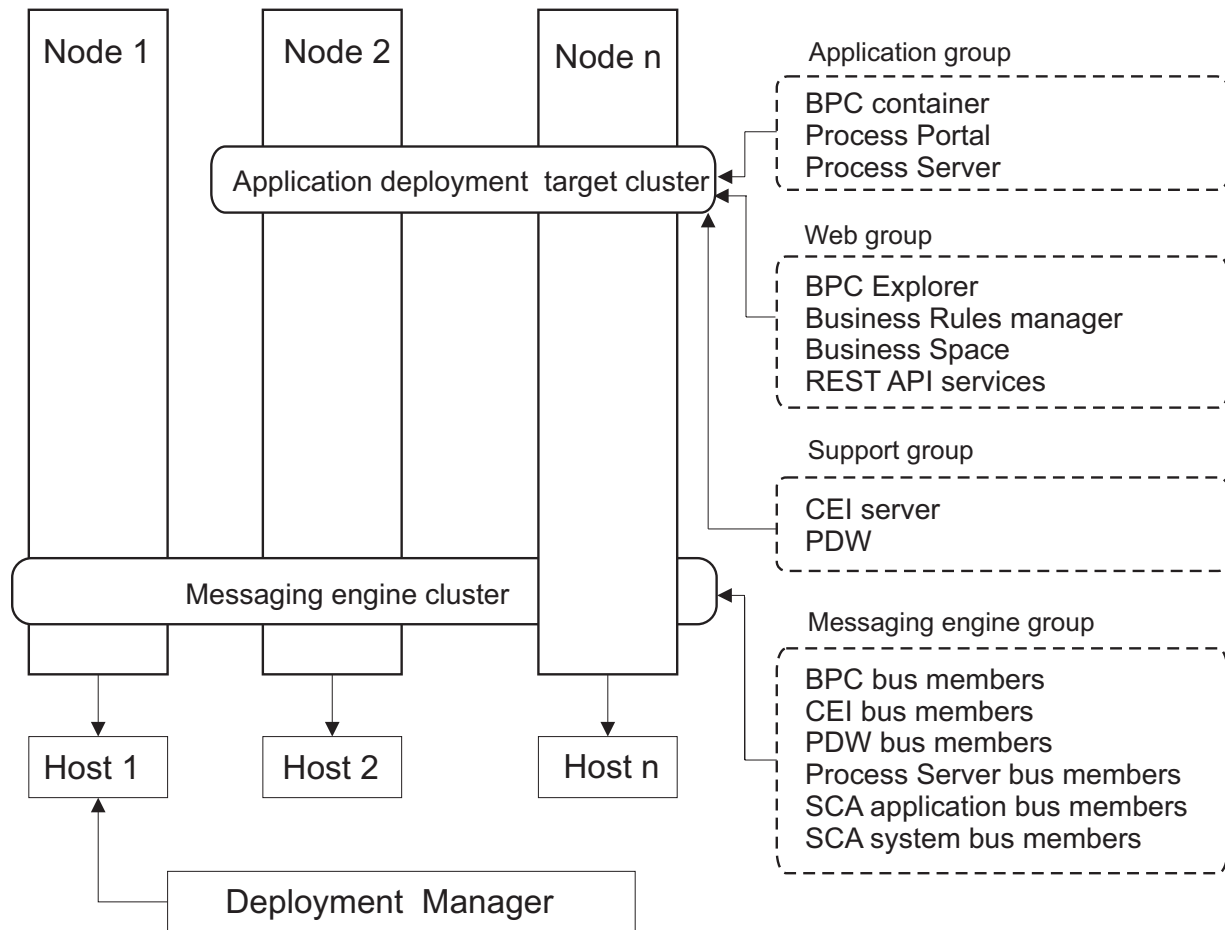


Figure 4. Remote Messaging topology pattern

Remote Messaging and Remote Support topology pattern

The *Remote Messaging and Remote Support* topology pattern is an IBM-supplied topology pattern. In a Remote Messaging and Remote Support topology pattern, the deployment environment functions are divided among three separate clusters.

With this three-cluster topology pattern components are divided among the messaging, support, and application deployment target clusters. For users of IBM Business Process Manager Standard, this topology pattern was the default and preferred topology. Now, for IBM Business Process Manager Standard the default and preferred topology is four clusters, with the additional web cluster hosting Business Space, Process Portal, and REST API services. For more information about the four-cluster topology, see Remote Messaging, Remote Support, and Web topology pattern.

When you create three clusters, each with its own functions and applications, you add an additional administrative burden. As you add clusters and cluster members, your performance tuning plan and the troubleshooting burden can expand greatly. Spreading messaging engines across the members of the messaging cluster also adds to the administrative burden associated with creating and maintaining policies.

From a scalability standpoint, the Remote Messaging and Remote Support topology pattern provides good flexibility. Because each of the distinct functions within IBM Business Process Manager is divided

among the three clusters, you can pinpoint performance bottlenecks and adjust the cluster size fairly easily. If you need additional Common Event Interface (CEI) processing, you can simply add a node and cluster member to the support cluster. Similarly, if you need more processing capability for your business processes or human tasks, you can add additional nodes and members to the application target cluster. Because expanding the messaging infrastructure beyond three cluster members has no effect on processing capability, the scalability limitations of the Remote Messaging topology pattern also apply to the Remote Messaging and Remote Support topology pattern.

Because the application target cluster runs your business integration applications only, performance tuning and diagnostics are simpler than in the topology patterns where the application target cluster has additional responsibilities. The Remote Messaging and Remote Support topology pattern is also ideal for environments that make extensive use of CEI for monitoring and auditing (including environments with IBM Business Monitor). When you separate the support infrastructure into its own cluster, you have a dedicated set of cluster members for CEI and for supporting applications.

The application deployment target cluster hosts the following:

- Business Process Choreographer (BPC) container
- Process Server
- Process Portal

The messaging infrastructure cluster hosts the following:

- Business Process Choreographer bus members
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members

The support infrastructure cluster hosts the following:

- The support applications:
 - CEI server application
 - Performance Data Warehouse
- The web applications:
 - Business Process Choreographer Explorer
 - Business Rules manager
 - Business Space
 - Process Portal
 - REST API Services

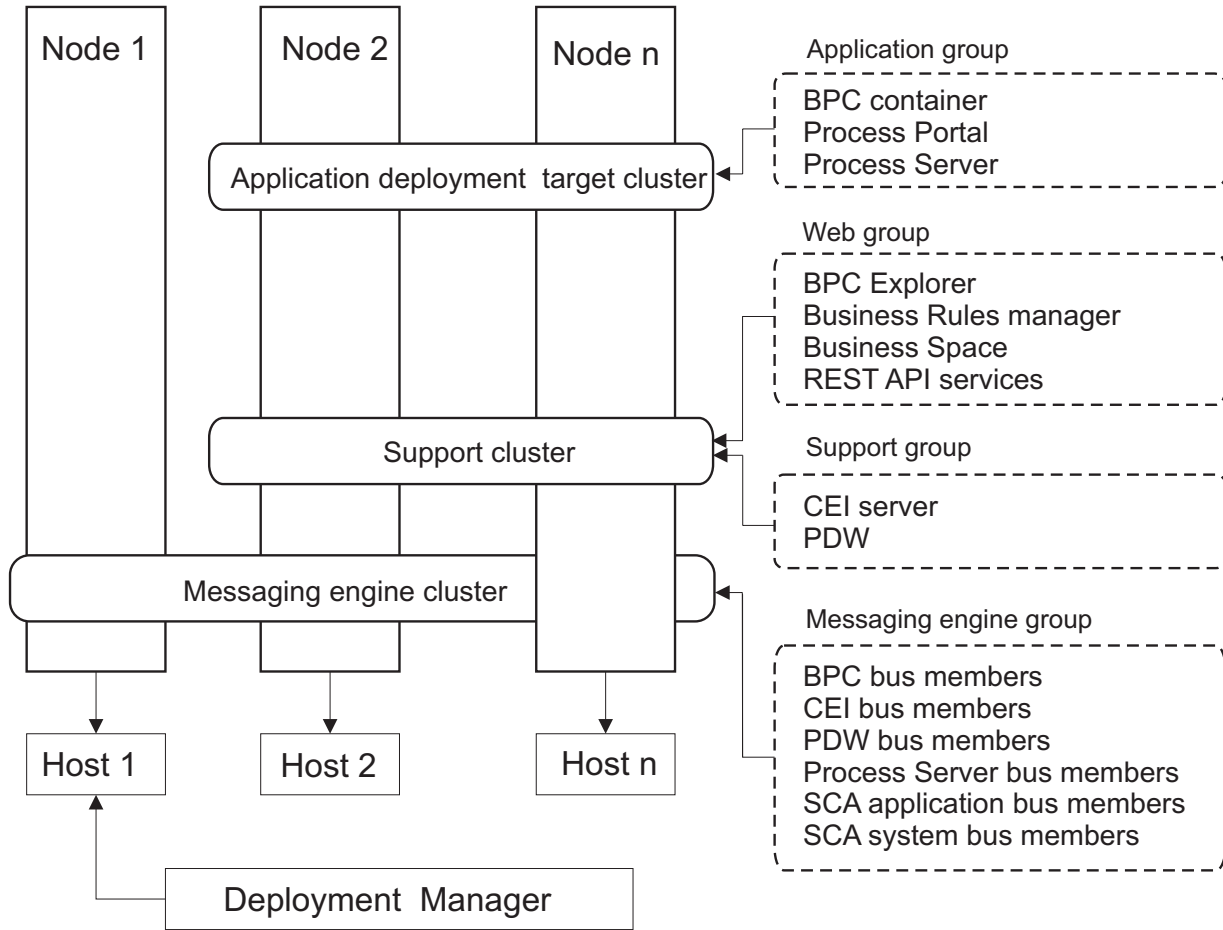


Figure 5. Remote Messaging and Remote Support topology pattern

Important: In this topology you must also configure a routing server such as IBM HTTP Server, WebSphere Application Server proxy server, or a reverse proxy server to ensure that requests that are intended for Process Portal are directed to the correct cluster.

Resource allocation example

The following figure shows one way to use the Remote Messaging and Remote Support topology pattern to allocate resources. The figure shows three hosts. Host A has Server 1 and Server 3; Host B has Server 2, Server 4, and Server 5 and Host C has Server 6 and Server 7. Because the heaviest load for this installation is for application use, more resources for Server 1, Server 2, and Server 6 are allocated for the application deployment target cluster (Cluster 3) than for the other functions.

Important: Load balancing is not available for the default configuration Remote Messaging and Remote Support topology pattern. That configuration uses a single messaging engine bus, while the load balancing feature requires at least two messaging engine buses.

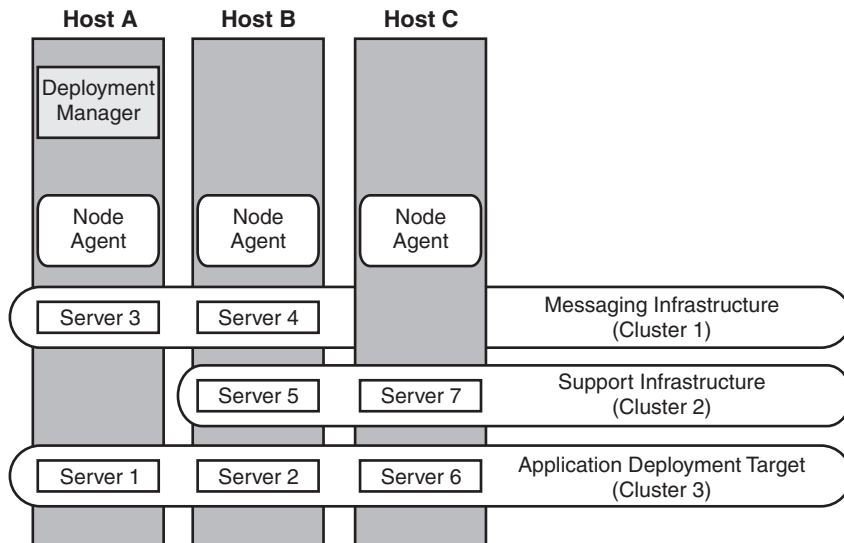


Figure 6. Resource allocation example

Remote Messaging, Remote Support, and Web topology pattern

The *Remote Messaging, Remote Support, and Web* topology pattern is an IBM-supplied topology pattern. In this pattern, the deployment environment functions are divided among four separate clusters.

The Remote Messaging, Remote Support, and Web topology pattern is the preferred topology for IBM Business Process Manager Standard and IBM Business Process Manager Advanced. It is also the default for the `configureNode` command, except for z/OS.

This four-cluster topology pattern is similar to the Remote Messaging and Remote Support topology pattern, except that supporting web applications reside on their own cluster.

The application deployment target cluster hosts the following:

- Business Process Choreographer (BPC) container
- Process Server
- Process Portal

The messaging infrastructure cluster hosts the following:

- Business Process Choreographer bus members
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members

The support infrastructure cluster hosts the following:

- CEI server application
- Performance Data Warehouse

The web application cluster hosts the following:

- Business Process Choreographer Explorer
- Business Rules manager
- Business Space
- Process Portal
- REST API Services

In a Remote Messaging, Remote Support, and Web topology pattern, the deployment environment functions are divided among four separate clusters. One cluster is used for messaging functionality, one cluster for support functionality, one cluster for applications, and one cluster for web-based functions. In addition to the ability to precisely control the individual components in your environment, the advantages of this topology pattern are similar to the advantages of the Remote Messaging and Remote Support topology pattern.

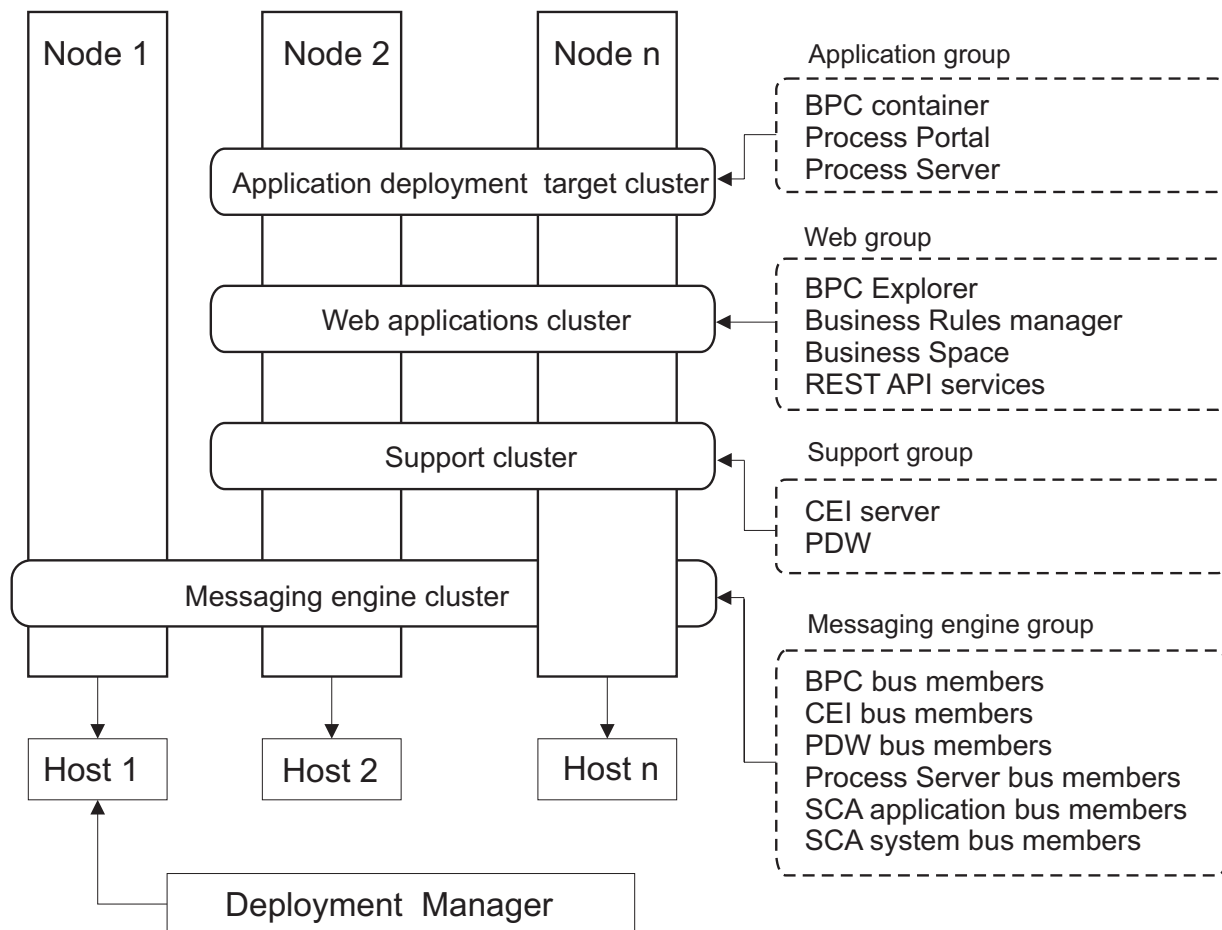


Figure 7. Remote Messaging, Remote Support, and Web topology pattern

Important: In this topology you must also configure a routing server such as IBM HTTP Server, WebSphere Application Server proxy server, or a reverse proxy server to ensure that requests that are intended for Process Portal are directed to the correct cluster.

Customized topology

A customized topology addresses the processing and business requirements unique to your situation. It is not an IBM-supplied topology pattern, but rather a topology that you create and then tailor to your specific needs.

If you need to define your own deployment environment topology, a customized topology is by far the most flexible. The IBM-supplied topologies (Single Cluster, Remote Messaging, Remote Messaging and Remote Support, and Remote Messaging, Remote Support, and Web), deploy all IBM Business Process Manager components to their default locations. You might not need the additional maintenance associated with these components. For example, if your organization does not need Common Event Interface (CEI), you could create a custom topology that removes CEI support from your environment. Similarly, if your organization has governance rules that prevent you from taking advantage of the Business Rules Manager, you could remove it from your deployment environment.

Except for the ability to control the components deployed in your environment, the advantages of custom topologies are similar to the advantages of the Remote Messaging and Remote Support topology. The disadvantages are also similar.

Important: Creating a customized network deployment environment is more labor-intensive than using an IBM-supplied topology pattern, which can be created from the Deployment Environment configuration wizard. Before you attempt to create a customized network deployment environment, make sure that none of the IBM-supplied topologies address your needs. You should attempt to create a customized network deployment environment only if you have a solid understanding of the features and functions of the administrative console.

Considerations for selecting a topology

Selecting an appropriate topology for your deployment environment depends upon several factors.

When you select a topology pattern, consider the following factors:

- Available hardware resources
- Application invocation patterns
- Types of business processes that you plan to implement (interruptible versus non-interruptible)
- How heavily you intend to use the Common Event Infrastructure (CEI)
- Individual scalability requirements
- Administrative effort involved

The Remote Messaging, Remote Support, and Web four-cluster topology pattern is the preferred topology for IBM Business Process Manager Standard and IBM Business Process Manager Advanced, but the choice ultimately depends upon your individual requirements. The four cluster topology is also the default for the **configureNode** command, except for z/OS. If you select a three or four cluster topology, you must configure a routing server for Business Space.

The IBM-supplied topologies can be applied to both Process Server and Process Center topologies. Therefore, your Process Center and Process Server network deployment environments can be organized in a similar way. The procedures for creating environments for Process Server and Process Center based on IBM-supplied topologies are also similar. The only difference related to IBM-supplied patterns is the recommended patterns for a production environment, and the components configured on the clusters for those patterns.

For information on the components, features, and functionality available in each of the IBM Business Process Manager configurations, see IBM Business Process Manager V8.0.1 configuration capabilities

Condensed topology pattern selection criteria

Consider the information listed in the following table, which is a quick guide to selecting your production topology. This table provides a condensed list of the advantages and disadvantages of each of the topology patterns.

For information about which BPM products support the supplied topology patterns, see *Topology patterns and supported BPM product features*.

Table 21. Considerations for selecting a topology for your deployment environment

Consideration	Topology Pattern			
	Single cluster	Remote Messaging	Remote Messaging and Remote Support	Remote Messaging, Remote Support, and Web
Number of clusters to maintain	One cluster for all components	Two clusters: One cluster for applications and for the support infrastructure One cluster for messaging	Three clusters: One cluster for applications One cluster for the support infrastructure One cluster for messaging	Four clusters: One cluster for applications One cluster for web interfaces One cluster for support infrastructure One cluster for messaging
Hardware requirements	Can be implemented on limited hardware	More hardware required for distributed environments	More hardware required for distributed environments	Most hardware intensive
Asynchronous interactions	Use should be minimal	Use must be balanced against resource availability	Ideal environment for asynchronous interactions	Ideal environment for asynchronous interactions
Long-running processes, state machines, and human tasks	Use should be minimal	Use must be balanced against resource availability	Ideal environment for interruptible processes, state machines, and human tasks	Ideal environment for interruptible processes, state machines, and human tasks
Heavy CEI activity	Not recommended (Light CEI use should be balanced against resource usage.)	Not recommended (Light CEI use should be balanced against resource usage.)	Ideal environment for heavy CEI use	Ideal environment for heavy CEI use
Administrative burden	Relatively small	Requires additional effort	Requires additional administrative effort	Requires most administrative effort
Scalability	All components scaled at the same rate	Messaging cluster scalability limited (no benefit beyond three servers) All other components scaled at the same rate	Easy to scale All functions separated Messaging cluster scalability still limited (no benefit beyond three servers)	Easiest to scale All functions separated Messaging cluster scalability still limited (benefit comes when other BPM products are introduced)
Routing server	Optional for load balancing and high availability	Optional for load balancing and high availability	Required for Process Portal – and can also be used for load balancing and high availability	Required for Process Portal – and can also be used for load balancing and high availability

The routing server can be IBM HTTP Server, WebSphere Application Server proxy server, or any other web server, reverse proxy server, or network layer IP sprayer.

Related concepts:

“Topology patterns and supported product features”

A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

“Load balancing and failover with IBM HTTP Server” on page 50

In a network deployment environment, you can configure IBM HTTP Server as a proxy server for workload balancing and failover purposes. Instead of incoming HTTP requests going directly to an application server, they go to the proxy server, which then distributes the requests across multiple application servers that perform the work.

“Topologies of a network deployment environment” on page 36

A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

Topology patterns and supported product features

A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

If you are using the Deployment Environment Configuration wizard on the administrative console to create the deployment environment, the availability of topology patterns on which you base your deployment environment varies depending on the following conditions and configuration decisions:

- The platform on which you have installed IBM Business Process Manager
- The primary deployment environment feature and the complimentary feature

Table 22 shows the relationship between the topology patterns and product features.

Table 22. Available supplied patterns and their relationship to product features

Topology pattern	Number of clusters	Description	Supported BPM products and features	Default status
Single Cluster	1	<p>Messaging, application deployment target, and application support functions are contained in a single cluster. This topology pattern is useful for synchronous messaging, proof of concept, or application testing environments.</p> <p>A Single Cluster topology pattern is ideal for limited hardware. Because all of the components are installed in the same cluster, fewer physical machines are required.</p>	<p>Supported by the following products, or any combination of these products:</p> <ul style="list-style-type: none"> • IBM Business Process Manager Advanced and Standard configurations • IBM Business Monitor 	

Table 22. Available supplied patterns and their relationship to product features (continued)

Topology pattern	Number of clusters	Description	Supported BPM products and features	Default status
Remote Messaging	2	<p>This topology pattern separates the messaging environment from the application deployment target and the application support functions. Use this topology pattern when message throughput is a critical requirement for your daily operation. This topology pattern is highly recommended for asynchronous messaging and transactional systems.</p> <p>In a Process Center network deployment environment, the Remote Messaging topology pattern is often enough to meet processing needs.</p>	<p>Supported by the following single product installations:</p> <ul style="list-style-type: none"> • IBM Business Process Manager Advanced and Standard configurations 	
Remote Messaging and Remote Support	3	<p>This topology pattern separates messaging, Common Event Infrastructure (CEI), application deployment target, and application support functions into distinct clusters. Most businesses can use this topology pattern to support their deployment environments because it is designed for performance and isolation of transactional processing from messaging and other support functions.</p>	<p>Supported by the following single product installations:</p> <ul style="list-style-type: none"> • IBM Business Process Manager Advanced and Standard configurations 	<p>This topology pattern is the default topology pattern for IBM Business Process Manager production environments.</p> <p>This topology pattern is the default topology pattern for the following installations:</p> <ul style="list-style-type: none"> • IBM Business Process Manager Advanced (including deployment environments) • IBM Business Process Manager Standard
Remote Messaging, Remote Support, and Web	4	<p>This topology pattern defines one cluster for application deployment, one remote cluster for the messaging infrastructure, one remote cluster for supporting applications, and one remote cluster for web application deployment (Business Process Choreographer Explorer, Business Rules Manager, Business Space, Process Portal, and REST API services).</p>	<p>Supported by the following products, or any combination of these products:</p> <ul style="list-style-type: none"> • IBM Business Process Manager Advanced and Standard configurations • IBM Business Monitor 	<p>This topology pattern is the default topology pattern for an IBM Business Monitor installation.</p> <p>For the Standard configuration, choose this pattern only if your topology includes IBM Business Monitor.</p>

Configurable components for each configuration

When you install the various configurations of IBM Business Process Manager, certain components are visible to you during the installation and configuration process. For network deployment, these components can be in one cluster or in multiple clusters.

Table 23 shows the components for each of the different configurations.

Table 23. IBM Business Process Manager components

Component	Adv	ESB	Std	Exp
Process Server	X		X	X
Process Server messaging engine (service integration bus)	X		X	X
Performance Data Warehouse	X		X	X
Performance Data Warehouse messaging engine (service integration bus)	X		X	X
Common database	X	X		
Common event infrastructure (CEI) messaging engine (service integration bus)	X	X		
Service Component Architecture (SCA)	X	X		
SCA application messaging engine (service integration bus)	X	X		
Business Space	X	X	X*	X*
Process Portal	X		X	X
Business Process Choreographer	X			
Business Process Choreographer messaging engine (service integration bus)	X			
Business Process Choreographer Explorer	X			

Note: On IBM BPM Standard and IBM BPM Express the Business Space configuration only supports Process Portal and Business Space spaces that include IBM Business Monitor widgets.

Related concepts:

“Considerations for selecting a topology” on page 45

Selecting an appropriate topology for your deployment environment depends upon several factors.

“Topologies of a network deployment environment” on page 36

A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

Determining whether to create a standardized or customized network deployment environment

There are two approaches to configuring your network deployment environment after profile creation. You can create a standardized network deployment environment based on IBM-supplied topology patterns, or you can create a customized network deployment environment to meet your unique business processing needs.

Reasons to create a standardized network deployment environment

If the IBM-supplied topology patterns (packaged as templates in IBM Business Process Manager) address all or most of your business processing needs, use the Deployment Environment wizard to create a

standardized network deployment environment. The Deployment Environment wizard generates clusters and servers according to several predefined topology patterns, and configures multiple components across them all at the same time.

You should choose to create a standardized network deployment environment for one or more of the following reasons:

- You do not have a lot of experience using the features and functions of the administrative console required to create and configure servers, server clusters, and IBM Business Process Manager components.
- You want to configure multiple components by using a single wizard in the administration application.
- You want to import a database design file to provide the values for database-related resource definitions.
- You have a predefined Deployment Environment that you can import into the current environment and customize, if necessary.

Reasons to create a customized network deployment environment

If the complexities of your business processing needs are not sufficiently met by any of the IBM-supplied topology patterns, use the features and functions of the administrative console to create a customized network deployment environment.

You might choose to create a customized network deployment environment for one or more of the following reasons:

- You are well-versed in using the administrative console to create deployment environments.
- You understand the concepts and component architecture required in a multi-clustered environment.
- You want to configure any clusters or servers upon which the components will be deployed *before configuring* the IBM Business Process Manager components themselves.

Load balancing and failover with IBM HTTP Server

In a network deployment environment, you can configure IBM HTTP Server as a proxy server for workload balancing and failover purposes. Instead of incoming HTTP requests going directly to an application server, they go to the proxy server, which then distributes the requests across multiple application servers that perform the work.

Tip: You do not have to use IBM HTTP Server for load balancing, you can use any other type of routing server, for example, WebSphere Application Server proxy server, any other web server, reverse proxy server, or network layer IP sprayer.

Although the specific configuration steps differ depending on your environment, the following general tasks can help you plan for and implement workload balancing and failover with IBM HTTP Server.

1. Review the information in *Selecting a front end for your WebSphere Application Server topology* to learn more about using a proxy server front end.
2. Implement a web server plug-in. Install and configure the plug-in as part of the local deployment manager profile. These WebSphere Application Server topics provide more information:
 - Implementing a web server plug-in
 - Installing and configuring web server plug-ins (refer to Scenario 5 in this topic)
3. Configure Secure Sockets Layer (SSL) between the deployment manager for WebSphere Application Server and the IBM HTTP Server administration server. You must configure the Application Server to accept a self-signed certificate from IBM HTTP Server so SSL connections are accepted and transactions are completed. See *Configure SSL between the IBM HTTP Server Administration Server and the deployment manager* for more information.

4. Customize the Process Server or Process Center cluster so that the configuration file points to the web server, the Process Server is in online mode, and communication is enabled for HTTP over SSL or HTTP Secure (HTTPS). See Customizing the Process Server or Process Center cluster to work with a web server.

Related concepts:

“Considerations for selecting a topology” on page 45

Selecting an appropriate topology for your deployment environment depends upon several factors.

Expanding your topology

Advanced

8.0.1.2+

There are three ways to expand your topology: adding members to a cluster, adding cells, or adding deployment environments.

Adding cluster members

The easiest way to expand your infrastructure is to add more cluster members to your existing clusters. You can add cluster members to each cluster independently or in combination, depending on where you see the need for growth. You can add an application cluster member, a messaging cluster member, or both, by adding a new node to the deployment environment. By expanding your clusters in this way, you can improve your application throughput.

Consider adding cluster members when one of the following situations occurs:

- You plan to deploy new applications to your existing environment.
- You anticipate increased volume requirements for your existing applications.
- You need more capacity for operational purposes, such as failover.

However, there are some cases where expanding existing clusters is not appropriate. Consider another solution in the following cases:

- You have applications or sets of applications that serve different business purposes and you want to keep them distinct. If you deploy these distinct applications to the same deployment environment, you might introduce dependencies between otherwise unrelated business domains. Dependencies can affect such things as maintenance schedules and application availability when, for example, unrelated applications are less reliable.
- After you analyze your performance characteristics, you realize that planned deployments might push your system beyond the limits of the current deployment target. They might introduce too many modules from new applications or new versions of applications to run in the existing memory space, or use your database tables too heavily.

Adding cells

If you decide that expanding your existing clusters is not an appropriate solution, consider creating another deployment environment in another cell. This approach gives you the most room for growth, the most flexibility for expanded functional requirements, and complete isolation for your applications.

To add another cell, you duplicate common cell-level configurations, such as global security settings. You use multiple consoles to manage your applications, such as separate administrative consoles and failed event managers.

Adding deployment environments

If the previous two options do not satisfy your requirements, there is a third alternative for expanding your topology. You can create two independent deployment environments for your applications in the same cell.

Consider adding a second deployment environment when one of the following situations occurs:

- You must expand the capacity of your environment, but adding cluster members does not satisfy your requirements.
- You want to isolate process applications into separate Java™ virtual machines (JVMs), associating each application group with a specific set of JVMs.
- You want to set up multiple test environments quickly without having to configure security or nodes each time.

The second deployment environment is for IBM Process Server only, not for IBM Process Center. You need one Process Center per cell, or the single repository experience is lost.

Running two deployment environments in the same cell is an advanced topology that requires research and planning.

Considerations for multiple deployment environments in the same cell

Advanced

8.0.1.2+

Before you implement this advanced topology, there are several important things to consider.

Maintenance considerations

Maintenance is more difficult with more than one deployment environment in the same cell.

- If there is an issue with one application in the cell, it is not possible to apply an interim fix only to the affected deployment environment. Interim fixes affect all servers, deployment environments, and clusters in the cell. Fixes for one application might have an unanticipated effect on the other applications that are running in the cell.
- Testing an interim fix from IBM is more difficult when several deployment environments are in the same cell. Separate cells help ensure that fixes do not break other applications.
- You might have to bring down all your servers to apply interim fixes for one set of clusters, resulting in downtime across all the sets of clusters using the cell. Although the exact arrangement of servers varies, a common arrangement of servers is to have one member of each cluster on each node. In such an arrangement, all servers and cluster members that share the node are affected by the steps to apply the interim fix.

Application considerations

Consider how applications are used with two different deployment environments in the same cell.

- You cannot install two instances of the same Service Component Architecture (SCA) application in the cell. You can install many SCA applications, but they must have different module names. You can use the **SCARename** utility to rename them.

As an example of this problem, if you create an environment with two deployment environments in the same cell and then try to import into both of them an IBM BPM export (.twx) file that contains BPEL processes, the second import fails because the SCA module names are the same and an SCA module name must be unique within a cell. To rename one of them, you must extract the EAR file from the TWX file, call the **SCARename** utility to rename it, and then package the new EAR in the TWX file.

There are several ways to manage this problem.

- You can use module naming conventions that incorporate the name of the target deployment environment.
- If you are deploying the modules as part of a process application in Process Center, you can use process app naming conventions that incorporate the name of the target deployment environment.
- If you are deploying directly as an EAR file, you can use module versioning where the version name uses a naming convention that incorporates the name of the target deployment environment.
- For late binding to work, new versions of a BPEL business process or human task (template) must be deployed to the same deployment environment as the earlier version. The correct target to bind to must be found in the same deployment environment. Make sure that parent-child relationships between processes or between human tasks are scoped to the deployment environment. There are some relationships, like parent-child flows, that should not cross JVMs.
- Each Process Portal has one view to each deployment environment and requires unique context roots. Consider whether to use a different web server for each deployment environment. If not, you must provide different virtual hosts to ensure unique context roots for applications.

Administration considerations

Consider how administration works with more than one deployment environment in the same cell.

- Each application cluster must have a corresponding support cluster and messaging engine cluster.
- Failed events from all deployment environments are stored in a single set of cell-scoped tables.
- You must ensure unique names for all applications that contain SCA modules such as BPEL processes, calendars, rules, selectors, and relationships.
- You must ensure unique names for IBM BPM applications as well as for customer applications.
- You must add databases and schemas for each set of clusters, which increases administration responsibilities. Each set of clusters requires databases and schemas for:
 - Process database
 - Performance Data Warehouse database
 - Common database at the deployment environment level

You do not need separate database instances in DB2. If you use one database, you must provide separate user IDs for each Process Server and Performance Data Warehouse. These user IDs are used as the schema and keep the tables separate.

Planning your database configuration

To plan your database configuration, you need to know which databases must be in place and configured in order to use the software, which components of IBM Business Process Manager you will use and their associated databases, the tasks required to administer the databases, and the security privileges of the database system that you are using.

Database server setup options

The Set up the Database Server page in the Installation launch pad offers varying options depending on your database server back-end.

Using an existing database or installing a new database

You can choose to have the installation process include automatically installing an instance of DB2 Express. Choose this option if you do not have an existing or planned supported database server you intend to use to store content from Process Center or Process Server. If you are installing a test or proof-of-concept environment, you would likely choose to have a new DB2 Express installed automatically, unless one already exists on the system. You must be installing as an administrator to have DB2 Express installed automatically.

Applying Windows authentication

Enable this option if you want to use an existing Windows user ID and password for SQL Server authentication rather than providing a user ID and password in the database options. For more information, see the SQL Server documentation: <http://technet.microsoft.com/en-us/library/ms144284.aspx>.

Specifying the Oracle instance name

If you are using an Oracle database server, provide the name of the Oracle instance to be configured. You can use a single instance of Oracle for configuring IBM BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different IBM BPM databases.

Databases and your IBM Business Process Manager topology

A database configuration is part of the overall IBM Business Process Manager topology. For configuration activities, you must understand the differences between Process Center and Process Server databases as they relate to topology and runtime processing.

Although Process Center shares the same database schema, there are specific differences in the instance data that is stored. You must have a separate instance of the shared schema for each instance of Process Center and Process Server.

If you use the **Typical profile creation** option, the database design tool automatically generates SQL scripts for a default database configuration.

If you create the stand-alone development profile (**qwps**) during installation, the required databases are configured automatically. You do not need to design the database requirements.

If you create database tables manually, use the database design tool to create the SQL scripts, because the tool ensures that the generated SQL scripts are unique.

Restriction: For Oracle, a database instance is not created as part of the SQL file execution; you must create an instance manually, using the Oracle product documentation.

You can incorporate configuration information for the database into the profile creation process by one of the following methods:

- Referencing a database design file
- Setting database configuration parameters with the Profile Management Tool or the **manageprofiles** command-line utility

Regardless of how you choose to implement your database configuration, you must generate the SQL scripts as part of the profile creation process. You cannot reuse SQL scripts between Process Servers or clusters of Process Servers.

Before they configure the databases, the solution architect and database administrator must collaborate on the database topology to understand the best way to store database tables. For example, will the tables be stored in the same database as the common database? Or will the tables be stored in a separate database as a stand-alone profile? Separate databases might be helpful because they simplify the database configuration. You might also use separate databases to tune and manage the component databases separately from the common database in a stand-alone server environment.

Process Center database and Process Server database

IBM Business Process Manager Express and IBM Business Process Manager Standard include database configuration patterns, so you can create a database design for each runtime environment. You can use either pattern to create a database configuration for Process Server and Performance Data Warehouse Server components.

IBM Business Process Manager Advanced includes the same database configuration patterns, plus an additional pattern called **bpm.advanced.standalone.dbDesign**. You can use the **bpm.advanced.standalone.dbDesign** pattern as the database schema for the Process Server and Performance Data Warehouse server components in a common database. Use this pattern when you are creating a stand-alone profile that combines the Process Server and Performance Data Warehouse runtime environments. If you use this pattern for a common database, a custom schema value is not specified.

You can also use the procsry and pctrsry database designs with the Deployment Environment wizard. For network deployment environments within the standard or advanced configurations, these patterns are part of the setup for the Process Server or Process Center on a deployment host.

Choosing how and when to configure the common database

You can create the required database tables either before or after configuring IBM Business Process Manager. The important thing to remember is that the databases (including their tables, schemas, and so on) must exist before the IBM Business Process Manager servers try to use them.

If you are planning to configure Business Process Choreographer, see Planning the topology, setup, and configuration path.

You can create the common database before, during, or after you create the IBM Business Process Manager profile.

- Before you configure IBM Business Process Manager:
 - Edit and run the default scripts that come with IBM Business Process Manager. You can use the default scripts to create only the common database and Business Process Choreographer tables.
 - Use the design file that was created using the database design tool (DDT).
- After you configure IBM Business Process Manager:
 - Use the Profile Management Tool to configure IBM Business Process Manager to work with the tables in the database as you create the profile. You can create and configure the database tables during profile creation, or delay creation and configuration until after the profile has been created. Use the Profile Management Tool to generate database scripts that you can use to create and configure the database tables. These generated scripts are ready to use. No editing is required.
 - Use the design file that you created using the database design tool (DDT).

Supported database types

Choosing a database depends on your operating system and on the features that you will use with IBM Business Process Manager.

See Table 24 on page 56 for a list of the databases that are supported with IBM Business Process Manager.

IBM Business Process Manager packages JDBC drivers for DB2, Oracle, and SQL Server. For information about the JDBC drivers (including version and level information), see the web page Detailed hardware and software requirements for IBM Business Process Manager.

Note: You are responsible for providing JDBC driver levels outside of what is packaged with IBM Business Process Manager.

The DB2 Express database is included in the IBM Business Process Manager software, and can be installed and configured automatically when you install IBM Business Process Manager. The user must have administrative privileges (root or Administrator) to install DB2 Express.

Important: Linux If you are installing DB2 Express as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. See Kernel parameter requirements (Linux) for a list of the kernel requirements. You can locate the current values by parsing the output of the `ipcs -l` command.

Note: Currently, there is a known limitation in DB2 Express installer related to the inclusion of national language (NL) strings in properties passed to it from the IBM Business Process Manager installer. The following values, which are passed to DB2 Express when it is being installed cannot have NL strings in them: Linux Windows

- Linux Instance user name and password: `bpminst`
- Linux Fenced user name and password: `bpmfenc`
- Linux Administration server (DAS) user name: `bpmadmin`
- Windows Administrative user name: `bpmadmin`

Choose passwords that comply with the password policy of your system (for example, Windows 2008).

Each database is represented by a parameter `dbType` which is a character string. The `dbType` parameter is used as a parameter in `manageprofiles` command-line utility. The values of `dbType` for the supported databases are shown in Table 24.

Table 24. Supported database types, their associated `dbType` values and restrictions

Supported database	dbType value	Restrictions and notes
DB2 Express	DB2_UNIVERSAL	Used as the default database type for a stand-alone profile.
DB2 Universal	DB2_UNIVERSAL	
DB2 Data Server	DB2_DATASERVER	Available for download from: 9.7 GA level Fix Packs
DB2 for z/OS	DB2UDBOS390	If you are using DB2 for z/OS as your database management system, you must configure the database and database objects by using the <code>createDB2.sh</code> script. The installation wizard is not able to create a database of this type. .

Table 24. Supported database types, their associated dbType values and restrictions (continued)

Supported database	dbType value	Restrictions and notes
Microsoft SQL Server	Microsoft SQL Server JDBC 1.2 and 2.0 = MSSQLSERVER_MICROSOFT	Microsoft SQL Server JDBC 3.0 is also supported, though not listed as a separate database type on the Profile Management Tool. If you are using Microsoft SQL Server JDBC 3.0 as your database management system, selecting a database type of Microsoft SQL Server JDBC 2.0 will support version 3.0 Note: The Process Server and Performance Data Warehouse databases cannot be shared in the CommonDB. Note: If a locale different from Latin must be specified, then the createDatabase.sql script must not be used. A different locale that is NOT case-sensitive must be specified. Important: You must configure XA transactions after the database is installed and before you start the server. Failure to configure the XA transactions can result in an error during server startup.
Oracle	ORACLE	The installation wizard is not able to create a database of this type for Oracle. Note: IBM BPM requires the national character set to be UTF8, based on the following recommendation from Oracle: The maximum column size allowed is 4000 characters when the national character set is UTF8 and 2000 when it is AL16UTF16. The maximum length of an NVARCHAR2 column in bytes is 4000. Both the byte limit and the character limit must be met, so the maximum number of characters that is actually allowed in an NVARCHAR2 column is the number of characters that can be written in 4000 bytes. For details, refer to the following Oracle information: http://www.oracle.com/technetwork/database/globalization/dmu/overview/index.html

A second parameter used in file path and file naming conventions is *feature*, which indicates which of the various databases is under consideration. Table 25 lists the databases and the associated *feature* parameter.

Table 25. Databases and their associated feature name.

Database	Feature
Business Process Choreographer	ProcessChoreographer
Business Space	BusinessSpace
Process Server	ProcessServer
Performance Data Warehouse	PerformanceDW
Common database	CommonDB
Enterprise service bus logger mediation database	EsbLoggerMediation

When you install IBM Business Process Manager, database scripts are created in the following locations:
INSTALL_ROOT/dbscripts/*feature*/*dbType*

Where *feature* can be:

- BusinessSpace
- CommonDB
- EsbLoggerMediation
- ProcessChoreographer
- ProcessServer
- PerformanceDW


The creation of most JDBC providers is performed during profile creation, based upon the database parameters that you provide. However, the Business Process Choreographer JDBC providers may be created later using scripts or the administrative console. If a suitable JDBC provider is located on the server or cluster, the existing JDBC provider is used and the creation of a new one is omitted.

Related concepts:

“JDBC drivers and locations” on page 59

The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

Related information:

 Using a DB2 database to hold custom user registry data

Database naming restrictions

Databases cannot be reused across multiple installations of IBM Business Process Manager. Each installation of IBM Business Process Manager requires exclusive use of its associated databases. You must configure the databases so that they can be uniquely identified.

Depending on the installation path that you select, the databases associated with an installation might be configured with default names. For example, the databases associated with IBM Business Process Manager Advanced on IBM DB2 have the following defaults:

- Process Server - BPMDB
- Performance Data Warehouse - PDWDB

If you have two installations of IBM Business Process Manager Advanced that use DB2, you must select, for one of the installations, an installation path that lets you specify the names instead of accepting the default values.

To ensure the uniqueness of the database names, select installation paths that prompt you for the database names.

For example, you are prompted for the database names when you use the Typical installation path. Select the option to use an existing database server instead of the default DB2 Express.

Note: When you use the Profile Management Tool to create a profile after installation, you are prompted for database names, no matter which path in the Profile Management Tool (Typical or Advanced) you choose. The only exception is when you use a database design file for your database configuration. The database design file contains the database names, user name, and password information. Thus, the Profile Management Tool does not prompt you for this information.

In contrast, you are not prompted for database names in the following cases:

- You are using the Custom installation path for the IBM Business Process Manager Advanced Process Server image. Installation Manager provides an option to create the profile, but assigns default values to the database names.

- You are using the Typical installation path and you choose to install DB2 Express, which is packaged with the installation images. Database names are assigned the default values.

Additional restrictions apply to database naming. These restrictions depend on the database server that you are using.

IBM Business Process Manager Advanced Advanced installation

For IBM DB2 and Microsoft SQL Server databases, the following restrictions apply:

- The Process Server, Performance Data Warehouse, and Common database components must be in separate databases.
- The Process Server and Performance Data Warehouse components do not support case-sensitive databases. These databases must NOT be case-sensitive.

For Microsoft SQL Server databases, components other than Process Server or Performance Data Warehouse require that their databases be case-sensitive.

For Oracle databases, the Process Server, Performance Data Warehouse, and Common database components must use a separate schema/user. They can use the same instance.

Database time zone and character set considerations

You should not change the time zone of the database server that you use for IBM Business Process Manager. The national character set required for IBM BPM is different from the Oracle default value.

Time zone considerations

Timestamps that are stored in the database are based on the time zone that is configured for the database server. Because of this dependency on the database time zone, you should not change the time zone configuration for the database server of an existing setup if timers are already scheduled. If you change the time zone configuration of the database server, the existing timers can trigger at an unexpected time.

If you have a second database server for failover or disaster recovery reasons, both database servers should be configured for the same time zone, regardless of where they are physically located.

Character set considerations for Oracle databases

IBM BPM requires that the Oracle database be Unicode enabled. In Oracle, there are two character set parameters that influence how CHAR and NCHAR types are stored in the database:

1. For database character set (CHAR types), IBM BPM requires AL32UTF8.
2. For national character set (NCHAR types), IBM BPM requires UTF8.

Note that the national character set required for IBM BPM is different from the Oracle default value.

For a more detailed description of the character set parameters, refer to the related Oracle information.

Related information:

Oracle Database Migration Assistant for Unicode

JDBC drivers and locations

The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

The following supported JDBC drivers are included with the product installation files.

Table 26. Supported JDBC drivers and locations that are provided with the product

Server	Driver description	Driver location	Comments
DB2	IBM DB2 Universal JDBC Driver 3.61.65	WAS_HOME/jdbcdrivers/DB2	IBM DB2 Universal JDBC Driver is the default DB2 driver for both distributed and z/OS platforms.
	IBM Data Server Driver for JDBC and SQLJ 4.11.69		
Oracle	Oracle JDBC Driver 11g 11.2.0.1.0	WAS_HOME/jdbcdrivers/Oracle	
SQL Server	Microsoft SQL Server JDBC Driver 2.0	WAS_HOME/jdbcdrivers/SQL Server	Microsoft SQL Server JDBC Driver 2.0 supports SQL Server 1.2, 2.0, and 3.0.

The following supported JDBC drivers are not included with the product installation files.

Table 27. Supported JDBC drivers that are not provided with the product

Server	Driver description
Oracle	Oracle JDBC Driver 11g 11.1.0.6
SQL Server	Microsoft SQL Server JDBC Driver 1.2 and 3.0

Related concepts:

“Supported database types” on page 55

Choosing a database depends on your operating system and on the features that you will use with IBM Business Process Manager.

Related information:



Detailed hardware and software requirements for IBM Business Process Manager Advanced



Detailed hardware and software requirements for IBM Business Process Manager Standard



Detailed hardware and software requirements for IBM Business Process Manager Express

Non-administrative user considerations

If you are installing IBM Business Process Manager as a non-administrative or non-root user and you want to create a test profile during installation, you must have the DB2 server installed before you begin the installation. Remember the database details so that you can enter them during the installation.

Tip: The Interactive Installation and Configuration Guide is a tool in this information center that you can use to generate a set of installation and configuration topics customized to your precise installation needs. You can save the generated output, print it, take it with you when you install on another computer, or share it with someone else on your team. Each time you need to install with other options, return to the form and generate a new guide, or use it to see what the installation and configuration process looks when you choose different options.

The considerations described in this topic apply to any install scenario where you choose to install using the **Typical** install option. Profiles are created automatically when you install using the **Typical** option.

Note: Test profiles apply only for use with an IBM BPM Advanced configuration.

To install as a non-administrative user, you have the following choices: Linux Windows UNIX

- Before installing the product, install a DB2 server separately. For information about installing DB2 as a non-administrative or non-root user, see Linux UNIX Windows

- **Linux** **UNIX** Non-root installation overview (Linux and UNIX)
- **Windows** Required user accounts for installation of DB2 server products (Windows)
- Logon as an administrator and use the product installer to install the DB2 server alone. Grant special permission to the non-administrative user. Then logon as the non-administrative user and install the product using the installed DB2 server.

Alternatively, instead of creating a test profile, you can create a profile after installation (this is the case for any install using the **Custom** install option). Use these steps:

1. Install the product without creating a profile. When you install as a non-administrative user, on the Install Packages page, you must clear the check box for DB2 Express. On Windows, if you have the option to install IBM Cognos® Business Intelligence, you must clear that check box as well.
2. On the Features page, expand the servers and make sure that none of the test profiles are selected.
3. Use the Profile Management Tool to create a stand-alone profile, or to create the deployment manager and the custom profiles. If you do not have a database installed, use the **Advanced** path for all. Do not use the **Typical** path. Select the option to delay the execution of the database scripts during profile creation.
4. If the databases were not created in advance, have the database administrator create the databases and tables after profile creation or augmentation.
5. For a network deployment:
 - a. Federate the custom profiles to the deployment manager.
 - b. Using the administrative console, create the required deployment environment

Database privileges

Set database privileges to determine the authority that you must have to create or access your data store tables for each supported database management system.

When you create schemas with the installer, Profile Management Tool, database design tool, or scripts, your user ID must have the authority to create tables. When the tables are created, you must have the authority to select, insert, update, and delete information in the tables.

The following table describes the database privileges that are needed to access the data stores.

Table 28. Database privileges

Database type	Minimum privileges required to create objects in the database	Minimum privileges required to access objects in the database
DB2	The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. The user ID needs system privileges CREATEDBA and CREATEDBC. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables.	The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures. Refer to Table 29 on page 63 for detailed DB2 database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components.

Table 28. Database privileges (continued)

Database type	Minimum privileges required to create objects in the database	Minimum privileges required to access objects in the database
DB2 for z/OS	The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. To create storage groups for the database, the user ID needs CREATESG, CREATEDBA, and CREATEDBC system privileges. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables.	The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures. Refer to Table 30 on page 64 for detailed DB2 for z/OS database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components.
Oracle	The user ID needs sufficient privilege to create relational tables and indexes in the data store schema. The database also needs a space quota in the default table space of the owner of that schema.	The user ID needs the SESSION privilege to connect to the database. If the same user ID owns both the data store schema, and the component that is connecting to the database, the user ID has sufficient privilege to manipulate the tables. Otherwise, the user ID needs SELECT, INSERT, UPDATE, ALTER, and DELETE object privileges on the tables that make up the data store, and the DROP ANY TABLE system privilege to enable the use of the TRUNCATE TABLE statement. The user ID also requires the CREATE INDEX privilege. You must create the Oracle database using a UTF-8 character set, which supports the other customer character sets that are supported by IBM Business Process Manager. See Table 31 on page 65 for detailed Oracle database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components.
SQL Server	The user ID ideally requires DB OWNER privileges on the data stores used for IBM Business Process Manager.	Configure the SQL Server for SQL Server and Windows authentication so that authentication to be based on an SQL server login ID and password. The user ID must be the owner of the tables, or a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. See Table 32 on page 69 for detailed SQL Server database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components.

In the following tables, the installation privileges are the privileges that are required to install and configure the product. The runtime privileges are the database privileges required to run the product.

Table 29 on page 63 describes additional DB2 database privileges for IBM Business Process Manager components.

Table 29. Detailed DB2 database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE
Business Space	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Messaging Engines	CREATE TABLE, CREATE INDEXTYPE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE Note: Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.
Process Server or Performance Data Warehouse	<p>Required to create the database:</p> <ul style="list-style-type: none"> • CREATEDBA • CREATEDBC <p>Required to populate the database with our schemas and stored procedures:</p> <ul style="list-style-type: none"> • CREATETAB • CREATEIN • DROPIN <p>Additional required privileges on the created tables:</p> <ul style="list-style-type: none"> • ALTER • DELETE • INDEX • INSERT • REFERENCES • SELECT • UPDATE 	<p>Required privileges on the tables in the Process Server and Performance Data Warehouse databases:</p> <ul style="list-style-type: none"> • DELETE • INSERT • REFERENCES • SELECT • UPDATE <p>The runtime user must have EXECUTE ON PROCEDURE privileges on the six stored procedures in the createProcedure_ProcessServer.sql script.</p> <p>The runtime user requires all of the listed privileges on the Performance Tracking Server database as well. In addition, the user also must be able to create new tables in the Performance Tracking database, requiring the CREATETAB privilege.</p> <p>To read the system metadata tables, the SELECT permission is required on syscat.tables, syscat.views, syscat.columns, syscat.tabconst, and sysibm.sysdummy1.</p>

Syntax for creating a database and assigning rights to a DB2 user:

```
create database database_name automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to database_name;
grant dbadm on database to user user_name;
UPDATE DB CFG FOR database_name USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR database_name USING LOGSECOND 64 DEFERRED;
connect reset;
```

Table 30 describes additional DB2 for z/OS database privileges for IBM Business Process Manager components.

Table 30. Detailed DB2 for z/OS database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE, USAGE
Business Space	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Messaging Engines	CREATE TABLE, CREATE INDEXTYPE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE Note: Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.
Process Server or Performance Data Warehouse	<p>Required to create the database:</p> <ul style="list-style-type: none"> • CREATESG • CREATEDBA • CREATEDBC <p>Required to populate the database with our schemas and stored procedures:</p> <ul style="list-style-type: none"> • CREATETS • CREATETAB • CREATEIN • DROPIN <p>Additional required privileges on the created tables:</p> <ul style="list-style-type: none"> • ALTER • DELETE • INDEX • INSERT • REFERENCES • SELECT • UPDATE 	<p>Required privileges on the tables in the Process Server and Performance Data Warehouse databases:</p> <ul style="list-style-type: none"> • DELETE • INSERT • REFERENCES • SELECT • UPDATE <p>The runtime user must have EXECUTE ON PROCEDURE privileges on the six stored procedures in the createProcedure_ProcessServer.sql script.</p> <p>The runtime user requires all of the listed privileges on the Performance Tracking Server database as well. In addition, the user also must be able to create new tables in the Performance Tracking database, requiring the CREATETS and CREATETAB privileges.</p> <p>To read the system metadata tables, the SELECT permission is required on sysibm.systables, sysibm.sysviews, sysibm.syscolumns, sysibm.syschecks, sysibm.sysrels, sysibm.systabconst, sysibm.systablespace, and sysibm.sysdummy1.</p>

Table 31 on page 65 describes additional Oracle database privileges for IBM Business Process Manager components.

Important: If you configure all the following components for a single Oracle database, you can create a superset of all the privileges that are specified for each component. If you configure the four components for numerous databases, you can set different privileges for each.

Table 31. Detailed Oracle database privileges

Component	Installation privileges	Runtime privileges
Common DB	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges: CREATE USER, ALTER USER, CREATE ANY TABLE, ALTER ANY TABLE, DROP ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE, INSERT ANY TABLE, DELETE ANY TABLE, CREATE ANY INDEX, DROP ANY INDEX, CREATE ANY VIEW, DROP ANY VIEW, CREATE ANY PROCEDURE, DROP ANY PROCEDURE, CREATE ANY SEQUENCE</p> <p>The following privileges are for the situation when a DBA has created a Common DB user and that user creates the database objects, owns them, and uses them at runtime: CREATE TABLE, CREATE VIEW, CREATE PROCEDURE, CREATE SEQUENCE</p>	<p>The following privileges are for the situation when a DBA has created a Common DB user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE PROCEDURE</p> <p>CREATE VIEW</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>
Business Space	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges: CREATE USER, ALTER USER, CREATE ANY TABLE, ALTER ANY TABLE, DROP ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE, INSERT ANY TABLE, DELETE ANY TABLE, CREATE ANY INDEX, DROP ANY INDEX, CREATE ANY VIEW, DROP ANY VIEW, CREATE ANY PROCEDURE, DROP ANY PROCEDURE</p> <p>The following privileges are for the situation when a DBA has created a Business Space user and that user creates the database objects, owns them, and uses them at runtime: CREATE TABLE, CREATE VIEW, CREATE PROCEDURE</p>	<p>The following privileges are for the situation when a DBA has created a Business Space user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE PROCEDURE</p> <p>CREATE VIEW</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>

Table 31. Detailed Oracle database privileges (continued)

Component	Installation privileges	Runtime privileges
<p>Business Process Choreographer</p>	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges: CREATE USER, ALTER USER, CREATE TABLESPACE, CREATE ANY TABLE, ALTER ANY TABLE, DROP ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE, INSERT ANY TABLE, DELETE ANY TABLE, CREATE ANY INDEX, DROP ANY INDEX, CREATE ANY VIEW, DROP ANY VIEW, CREATE ANY PROCEDURE, DROP ANY PROCEDURE</p> <p>The following privileges are for the situation when a DBA has created a Business Process Choreographer user and that user creates the database objects, owns them, and uses them at runtime: CREATE TABLE, CREATE VIEW, CREATE PROCEDURE.</p>	<p>The following privileges are for the situation when a DBA has created a Business Process Choreographer user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE PROCEDURE</p> <p>CREATE VIEW</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>
<p>Common Event Infrastructure (CEI)</p>	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges: CREATE USER, ALTER USER, CREATE INDEXTYPE, ALTER ANY TABLE, ALTER SESSION, CREATE TABLESPACE, CREATE PROFILE, CREATE ROLE, CREATE TEMPORARY TABLESPACE</p> <p>The following privileges are for the situation when a DBA has created a Common Event Infrastructure user and that user creates the database objects, owns them, and uses them at runtime: CREATE TABLE, CREATE VIEW, CREATE PROCEDURE.</p>	<p>The following privileges are for the situation when a DBA has created a Common Event Infrastructure user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE PROCEDURE</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>

Table 31. Detailed Oracle database privileges (continued)

Component	Installation privileges	Runtime privileges
Messaging Engines	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges: CREATE USER, ALTER USER, CREATE ANY TABLE, ALTER ANY TABLE, DROP ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE, INSERT ANY TABLE, DELETE ANY TABLE, CREATE ANY INDEX, DROP ANY INDEX</p> <p>The following privileges are for the situation when a DBA has created a Messaging Engines user and that user creates the database objects, owns them, and uses them at runtime: CREATE TABLE</p>	<p>The following privileges are for the situation when a DBA has created a Messaging Engines user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>DROP ANY TABLE</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p> <p>Tip: Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. See Database privileges. If you prefer not to give the user this privilege, you can avoid it, at a performance cost, using tuning. See Setting tuning properties.</p>

Table 31. Detailed Oracle database privileges (continued)

Component	Installation privileges	Runtime privileges
<p>Process Server</p>	<p>The following privileges are for the situation when a DBA creates users and all the database objects for those users. The DBA requires the following privileges:</p> <p>CREATE USER</p> <p>ALTER USER</p> <p>CREATE TABLESPACE</p> <p>CREATE ANY TABLE</p> <p>ALTER ANY TABLE</p> <p>DROP ANY TABLE</p> <p>SELECT ANY TABLE</p> <p>UPDATE ANY TABLE</p> <p>INSERT ANY TABLE</p> <p>DELETE ANY TABLE</p> <p>LOCK ANY TABLE</p> <p>CREATE ANY INDEX</p> <p>DROP ANY INDEX</p> <p>CREATE ANY VIEW</p> <p>DROP ANY VIEW</p> <p>CREATE ANY PROCEDURE</p> <p>DROP ANY PROCEDURE</p> <p>CREATE ANY SEQUENCE</p> <p>DROP ANY SEQUENCE</p> <p>The following privileges are for the situation when a DBA has created a Process Server user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE TABLE</p> <p>CREATE PROCEDURE</p> <p>CREATE SEQUENCE</p> <p>CREATE VIEW</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>	<p>The following privileges are for the situation when a DBA created a Process Server user and that user creates the database objects, owns them, and uses them at runtime.</p> <p>explicit privileges:</p> <p>CREATE TABLE</p> <p>CREATE PROCEDURE</p> <p>CREATE SEQUENCE</p> <p>CREATE VIEW</p> <p>implicit privileges (included in the ownership of the database objects):</p> <p>SELECT</p> <p>INSERT</p> <p>UPDATE</p> <p>DELETE</p>

Syntax for assigning rights to an Oracle user:

```
CREATE USER user_name IDENTIFIED BY user_password;
GRANT connect, resource, unlimited tablespace TO user_name;
GRANT create view TO user_name;
GRANT javauserpriv TO user_name;
GRANT execute on dbms_lock TO user_name;
```

Example for creating a table space:

```
CREATE TABLESPACE user_table_space DATAFILE '/disk6/user_table_space.dbf'
SIZE 200M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING;
CREATE USER DB_user_name IDENTIFIED BY user_password DEFAULT user_table_space;
```

Table 32 describes additional SQL Server database privileges for IBM Business Process Manager components.

The IBM Business Process Manager database user should be assigned to the following three roles:

Note: The database must be created by the database administrator who can then assign these roles to the database user for IBM Business Process Manager.

- db_ddladmin
- db_datawriter
- db_datareader

For information regarding the permissions that are provided by these roles, refer to Microsoft-specific documentation.

The following table describes the permissions required of each component for both install and runtime activities.

Table 32. Detailed SQL Server database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, ALTER TABLE, INSERT, CREATE USER, ALTER USER	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Space	CREATE TABLE, ALTER TABLE, INSERT, CREATE USER, ALTER USER	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Messaging Engines	CREATE TABLE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE Note: Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.
Process Server	CREATE TABLE, SELECT, INSERT, UPDATE, DELETE TABLE, DROP TABLE, CREATE INDEX, CREATE VIEW, DROP VIEW, CREATE PROCEDURE, CREATE USER, ALTER USER	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, DROP TABLE, DROP VIEW

Table 32. Detailed SQL Server database privileges (continued)

Component	Installation privileges	Runtime privileges
Performance Data Warehouse	CREATE TABLE, SELECT, INSERT, UPDATE, DELETE TABLE, DROP TABLE, CREATE INDEX, CREATE VIEW, DROP VIEW, CREATE PROCEDURE, CREATE USER, ALTER USER	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, DROP TABLE, DROP VIEW

Syntax for assigning rights to an SQL user and creating the schema for the PDWDB database:

```
USE database_name
GO
CREATE USER user_name FOR LOGIN user_name
GO
CREATE SCHEMA user_name AUTHORIZATION user_name
GO
ALTER USER user_name WITH DEFAULT_SCHEMA=user_name
```

Important: For Performance Data Warehouse users, note the following restrictions:

- The user cannot be assigned to the system administrator (sysadmin) role.
- The user must be mapped to the PDWDB database with the db_owner role.
- The user must be mapped to the master database with the SqlJDBCXAuser role.

For more information, see the WebSphere Application Server page in the related reference.

Related reference:

 Database privileges in WebSphere Application Server

User ID or schema name privileges

During the installation of IBM Business Process Manager, you can use the default schema name and user ID privileges to install your databases. However, your database design might require separate user ID or schema name privileges.

Review the provided scenarios to determine when and how to configure different schema names and user ID privileges when you install IBM Business Process Manager.

Scenario for a single user ID or schema name privileges

If you chose a default installation for your databases, IBM Business Process Manager requires a minimum of one user ID or schema name that can create tables and to select, insert, update, and delete rows in those tables. You can use the Profile Management Tool or the installer to create your databases.

The following table shows the default database configuration properties when you use DB2 as your database. Other databases have different default configuration properties for database configuration.

Table 33. Scenario: Single user ID or schema

Database tables	Default database name with DB2	User ID or schema name
Common database tables	CMNDB	IBM Business Process Manager provides a user ID during installation.
Business Process Choreographer	BPEDB	IBM Business Process Manager provides a user ID during installation.

Table 33. Scenario: Single user ID or schema (continued)

Database tables	Default database name with DB2	User ID or schema name
Messaging tables	MEDB	IBM Business Process Manager provides a schema name during installation.

If your database design has different properties, you might need multiple user ID and schema name privileges. The following scenarios show you how to apply the configuration to achieve your design. Even if your particular design is not in the provided scenarios, you can adapt some of the ideas to implement your particular design.

Scenario 1 for multiple user ID or schema name privileges

In this scenario, you use a schema name that is the same as the user ID privileges, but you do not use the default schema name or default user ID privileges. This single user ID can access all of the database and also create all needed tables. The following examples show scenario 1 privileges:

- Schema name: dog
- Schema name for SCA.SYSTEM ME : dogSYS
- Schema name for SCA.APP ME: dogAPP
- Schema name for Event ME: dogEvent
- Schema name for BPC ME: dogBPC
- User ID to create schemas: dog
- User ID to select, insert, update, and delete schemas: dog

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 34. Scenario 1: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation 	This schema name is the same as the user ID that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation

Table 34. Scenario 1: Multiple user ID or schema (continued)

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Business Process Choreographer tables	<p>You supply this value twice:</p> <ol style="list-style-type: none"> In table creation scripts While configuring a deployment target using one of the following: <ul style="list-style-type: none"> Administrative console Installation wizard bpeconfig.jacl 	This schema name is the same as the user ID that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	<p>You supply this value twice:</p> <ol style="list-style-type: none"> In table creation scripts While configuring a deployment target using one of the following: <ul style="list-style-type: none"> Administrative console Installation wizard bpeconfig.jacl

Scenario 2 for multiple user ID or schema name privileges

In this scenario, you use the same schema name and user ID to select, insert, update, and delete schemas. However, you use a different user ID to create the schemas. The following examples show scenario 2 privileges:

- Schema name: snow
- Schema name for SCA.SYSTEM ME: snowSYS
- Schema name for SCA.APP ME: snowAPP
- Schema name for Event ME: snowEvent
- Schema name for BPC ME: snowBPC
- User ID to create the schemas: rock
- User ID to select, insert, update, and delete schemas: snow

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 35. Scenario 2: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	<p>You supply this value twice:</p> <ol style="list-style-type: none"> In table creation scripts During the IBM Business Process Manager configuration with one of the following: <ul style="list-style-type: none"> Administrative console Installation wizard Profile Management Tool Silent install Silent profile creation bpeconfig.jacl <p>Restriction: If you run the installer first, then you supply the value once because the generated scripts already contain the correct schema name and user ID values.</p>	The table creation scripts need to be modified with the schema name that allows reading and writing rows.	The table creation script needs to be modified with the user ID that allows table creation.	<p>You supply the user ID during profile creation through one of the following:</p> <ul style="list-style-type: none"> Installation wizard Profile Management Tool Silent install Silent profile creation
Business Process Choreographer tables	<p>You supply this value twice:</p> <ol style="list-style-type: none"> In table creation scripts While configuring a deployment target using one of the following: <ul style="list-style-type: none"> Administrative console Installation wizard bpeconfig.jacl 	The table creation scripts need to be modified with the schema name that allows reading and writing rows.	The table creation script needs to be modified with the user ID that allows table creation.	<p>You supply the user ID during profile creation through one of the following:</p> <ul style="list-style-type: none"> Installation wizard Profile Management Tool Silent install Silent profile creation

Scenario 3 for multiple user ID or schema name privileges

In this scenario, you use the same user ID to create all schemas. However, each schema has a different user ID to select, insert, update, and delete rows. The following list shows examples of privileges for Scenario 3:

- Schema name: waterCom
- Schema name for common tables: waterCom
- Schema name for SCA.SYSTEM ME: waterSYSME
- Schema name for SCA.APP ME: waterAPPME

- Schema name for Event ME: waterEventME
- Schema name for BPC ME: waterBPCME
- Schema name for BPC and HTM tables: waterBPC
- Schema name for ESBMessaging tables: waterESB
- User ID to create schemas: milk
- User ID to select, insert, update, and delete schemas:

Schema name	User ID to select, insert, update, and delete schemas
waterCom	waterCom
waterSYSME	waterSYSME
waterAPPME	waterAPPME
waterEventME	waterEventME
waterBPCME	waterBPCME
waterBPC	waterBPC
waterESB	waterESB

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 36. Scenario 3: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation 	This schema name is the same as the user ID that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply the user ID during profile creation through one of the following: <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation
Business Process Choreographer tables	You supply this value twice: <ol style="list-style-type: none"> 1. In table creation scripts 2. While configuring a deployment target using one of the following: <ul style="list-style-type: none"> • Administrative console • Installation wizard • bpeconfig.jacl 	The table creation scripts need to be modified with a schema name that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply this value twice: <ol style="list-style-type: none"> 1. In table creation scripts 2. While configuring a deployment target using one of the following: <ul style="list-style-type: none"> • Administrative console • Installation wizard • bpeconfig.jacl

Table 36. Scenario 3: Multiple user ID or schema (continued)

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Messaging tables	You supply this value with the definition of each messaging engine.	The table creation scripts must include the schema name that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply this value during the creation of the messaging engine. Select the Create Table option during the messaging engine configuration.

Planning your component-specific database configurations

IBM Business Process Manager includes components that require database tables and specific names of the databases where the tables are stored.

Use the information in this section to familiarize yourself with IBM Business Process Manager components that your database administrator must manage, configure, and administer.

To plan your database configuration, you must know the components that you will use. Table 37 lists the IBM Business Process Manager components that require a database table, and the default names of the databases where the tables associated with these components are stored.

Important: You can change these names if you choose, but you must remember to use the names consistently in later configuration steps.

Table 37. Databases that are required by individual components

Server component	Database (default name)	Notes
Process Server	BPMDB	The Process Server tables are used to hold, store, and track the data for the process application repository.
Performance Data Warehouse	PDWDB	
Application scheduler	CMNDB (the common database)	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables either during the startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server.
Business Process Choreographer	BPEDB	You must create the BPEDB before you start a server or cluster with Business Process Choreographer configured.
Business Process Choreographer Explorer reporting function	OBSRVDB	The Business Process Choreographer Explorer reporting functionality is deprecated. To monitor and report on BPEL processes, use IBM Business Monitor.

Table 37. Databases that are required by individual components (continued)

Server component	Database (default name)	Notes
Business Space	CMNDB (the common database)	For stand-alone profiles, you must create the common database before you start IBM Business Process Manager. For other profiles, you must use the administrative console to configure Business Space. Configuring a Business Space database is mandatory for using Business Space powered by WebSphere and Process Portal. Business Space powered by WebSphere provides a common interface for application users to create, manage, and integrate web interfaces across a range of IBM products.
Common Event Infrastructure (CEI)	EVENT (stores events)	CEI database configuration is not supported by the Profile Management Tool or the manageprofiles command-line utility. Do not create this database for production environments because the performance of persisting events may be impacted. The Common Base Event browser relies on the CEI database. If you want to use the Common Base Event browser to retrieve and view logging, tracing, management, and business events in your business enterprise applications, you must create the CEI database manually.
Enterprise service bus	CMNDB (the common database)	You must configure these tables either during startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server
Mediation	CMNDB (the common database)	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables either during the startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server.
Recovery	CMNDB (the common database)	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables either during the startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server.

Table 37. Databases that are required by individual components (continued)

Server component	Database (default name)	Notes
Relationships	CMNDB (the common database)	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables before or during the startup of the deployment manager or stand-alone server.
Selectors and business rules	CMNDB (the common database)/Repository DB	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables before or during the startup of the deployment manager or stand-alone server.
SIBus	User created	You must configure these tables during the startup of the messaging engine or before you start the messaging engine. You can use a file store with SIBus in a stand-alone environment during profile creation. However, you cannot use a file store with SIBus in a network deployment environment.

Table and schema creation

Depending on your database provider, you need different database tables and schemas for your database in IBM Business Process Manager.

In the administrative console, when you select **Resources > JDBC > Business Integration Data Sources** and click a data source, you can select the check box for **Create Tables** to create the tables from the data source. If only a database administrator can create database tables, clear the check box, locate the scripts in the message box, and give these scripts to your database administrator to run. If you enable **Create Tables**, various deployment environment functions of each database provider are available, as shown in the following tables. An “X” indicates that the table or schema is created.

Tip: You configure the common database during profile creation.

Table 38. Table creation based on database provider

Database providers	Messaging Engine	Business Process Choreographer	Business Process Choreographer reporting	IBM Process Server	Performance Data Warehouse
DB2	X	X	X	X	X
Oracle	X	X	X	X	X
Microsoft SQL	X	X		X	X

Table 39. Schema creation based on database provider

Database providers	Messaging Engine	Business Process Choreographer	Business Process Reporting	IBM Process Server	Performance Data Warehouse
DB2	X	X	X	X	
DB2 for z/OS	X	X	X	X	X

Table 39. Schema creation based on database provider (continued)

Database providers	Messaging Engine	Business Process Choreographer	Business Process Choreographer Reporting	IBM Process Server	Performance Data Warehouse
Oracle	X	X	X	X	

The only way to configure the Common Event Infrastructure database is manually. For more information, see *Configuring the Common Event Infrastructure database*.

Planning to configure the common database

The common database configurations contain information about supported database types, script names and their locations, profile creation configuration actions, installation parameters, types of created tables, and user ID privileges.

The common database acts as a repository for various components. The IBM Business Process Manager common database is used by the following product components:

- Business Space
- Recovery
- Relationship service
- Mediation
- Application Scheduler
- Customization (selector and business rule group)
- EventSequencing (LockManager)
- Enterprise Service Bus (ESB) Logger Mediation Primitive

You can create the common database before, during, or after you create the IBM Business Process Manager profile.

- Before you configure IBM Business Process Manager:
 - Edit and run the default scripts that come with IBM Business Process Manager. You can use the default scripts to create only the common database and Business Process Choreographer tables.
 - Use the design file that was created using the database design tool (DDT).
- After you configure IBM Business Process Manager:
 - Use the Profile Management Tool to configure IBM Business Process Manager to work with the tables in the database as you create the profile. You can create and configure the database tables during profile creation, or delay creation and configuration until after the profile has been created. Use the Profile Management Tool to generate database scripts that you can use to create and configure the database tables. These generated scripts are ready to use. No editing is required.
 - Use the design file that you created using the database design tool (DDT).

Supported database types

The common database can use the following database products:

Table 40. Supported database products

Database Types	Considerations
DB2 Express	Used as the default database type for a stand-alone profile.
DB2 Universal	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.

Table 40. Supported database products (continued)

Database Types	Considerations
DB2 Data Server	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 for z/OS v8 DB2 for z/OS v9	Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
Microsoft SQL Server (Microsoft)	
Oracle	You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas.

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see Database privileges and User ID or schema name privileges.

Database Management Service instances

For a network deployment environment, there is one set of common database tables per cell.

Configuration actions during profile creation

You can use one of the following options to install the common database:

- Installer
- Profile Management Tool
- Silent installation
- Scripts

Within each of these options are several more choices.

Installer

Use the Installer if you are going to create your profiles when you install your software. You can install your database products during installation, but you are limited in the types of database products that you can install. To use a supported database product that is not in Table 41, you must use the Profile Management Tool to create your deployment manager.

Table 41. Installer options

Option	Databases you can use
Typical: stand-alone profile	DB2 Express
Typical: all other profiles	DB2 Express
Customized: stand-alone profile	<ul style="list-style-type: none"> • DB2 Universal • DB2 Data Server • Oracle

Table 41. Installer options (continued)

Option	Databases you can use
Customized: all other profiles	<ul style="list-style-type: none"> • DB2 Universal • DB2 Data Server • Oracle

A typical installation uses default values for configuration parameters and you cannot change these defaults. If you choose a customized installation, you can change the defaults for your specific requirements.

Profile Management Tool

Use the Profile Management Tool to create profiles after you install your software. The Profile Management Tool allows you the options of installing your database before, during or after profile creation.

Table 42 lists the databases supported by typical (default) profiles and customized profiles.

Table 42. Profile Management Tool options

Options	Databases you can use
Typical: stand-alone profile	<ul style="list-style-type: none"> • DB2 DataServer • DB2 Universal • DB2 for z/OS • Microsoft SQL Server • Oracle <p>DB2 for z/OS, Oracle and Microsoft SQL databases must exist so that the Profile Management Tool can configure them.</p>
Typical: all other profiles	<ul style="list-style-type: none"> • DB2 DataServer • DB2 Universal • DB2 for z/OS • Microsoft SQL Server • Oracle <p>DB2 for z/OS, Oracle and Microsoft SQL databases must exist so that the Profile Management Tool can configure them.</p>
Customized: stand-alone profile	Any of the "Supported database types" on page 78
Customized: all other profiles	Any of the "Supported database types" on page 78

Silent Installations

When you install the product silently, you can specify the common database configuration by editing the template response file.

Scripts

You can use scripts to create your common database before you install IBM Business Process Manager or during profile creation.

If you choose to configure your database manually after profile creation, you must first install IBM Business Process Manager and indicate in the Profile Management Tool that you do not want to run the scripts as part of profile creation. The Profile Management Tool updates the default scripts with the database parameters that you specify, and writes updated scripts out to the *profile_root/dbscripts/CommonDB/dbType/dbName* directory.

Tip: You can use the Profile Management Tool to change the directory to which updated scripts are written.

The scripts are ready to run, but you can edit them to include any specific requirements. You can then give these scripts to the person who should create your common database. If you try to start IBM Business Process Manager before creating the database, you receive an error message.

When you run the scripts, you also perform the following tasks:

- Create a database, if appropriate (valid only for a local database), depending on your choices in the Database configuration panel in the Profile Management Tool.

Important: Although you can defer creating the database until the profile creation is complete, you must enter valid information in the Database configuration panel of the Profile Management Tool. This information is used to create the data source for the IBM Business Process Manager.

- Create the data source on the JDBC provider.

Important: If you create the database at the same time as the profile, and if you introduce mistakes in the database parameters, errors occur in the profile. If you delay creating the database, the profile is created without errors, but the generated database scripts contain errors and you must correct them before you can create the database. For custom (managed) nodes of a cell, you must select the same database type as the deployment manager profile. The data source is maintained only at the cell level.

Note: For custom (managed) nodes of a cell, you must select the same database type as the deployment manager profile. The data source is maintained only at the cell level.

SQL scripts

Use SQL scripts to configure your database before or after you create the profile. Tables are created with a deployment manager profile so no SQL scripts are run while the managed node is created.

You can find SQL scripts for each common database client in the following location:

- *install_root/dbscripts/CommonDB/dbType* after you install IBM Business Process Manager

If you choose to defer creation of the database after you create the profile, you can find the updated scripts in the *profile_root/dbscripts/feature/dbType/dbName* directory.

The SQL script naming convention is:

- For a component-specific script: **createTable_componentName.sql**, for example **createTable_Recovery.sql**
- For a component-independent script: **createTable.sql**.

The following table shows the script naming convention.

Table 43. Common database script naming convention

Type of script	Script name
Component specific	<i>scriptName_componentName.sql</i>
Component independent	<i>scriptName.sql</i>

JDBC provider

A new Java Database Connectivity (JDBC) provider is created depending on the database type. The provider is created in the node scope in a stand-alone profile and at the cell level in a network

deployment environment. The JDBC provider refers to the `JDBC_DRIVER_PATH` variable to locate local JDBC drivers. The variable is specified at the cell level and each node level points to the correct local path.

Data source name:

- **WPS DataSource**

Data source JNDI name:

- `jdbc/WPSDB`

Restrictions

Several restrictions exist for the database commands that are available during profile creation.

Create new database is disabled for the following database types:

- DB2 for z/OS
- Oracle
- Microsoft SQL Server

Tables

The common database scripts create only static tables during profile creation. The following table contains a list of all the tables that are created by different components.

Table 44. Tables created by IBM Business Process Manager components

Component	Table names	Scripts
Recovery	FAILEDEVENTS FAILEDEVENTBOTYPES FAILEDEVENTMESSAGE	<code>createTable_Recovery.sql</code>
Mediation	MEDIATION_TICKETS	<code>createTable_mediation.sql</code>
Relationship	Dynamic table, created at run time	<code>createTable_Relationship MetadataTable.sql</code>
Application Scheduler	WSCH_LMGR WSCH_ LMPR WSCH_TASK WSCH_TREG	<code>createTable_AppScheduler.sql</code>
Customization (selector/business rule group)	BYTESTORE BYTESTOREOVERFLOW APPTIMESTAMP	<code>createTable_customization.sql</code>
Common database	SchemaVersionInfo	<code>createTable_CommonDB.sql</code>
Persistent LockManager	PERSISTENTLOCK	<code>createTable_lockmanager.sql</code>
ESB Logger Mediation	MSGLOG	<code>createTable_ESBLogger Mediation.sql</code>

All the SQL scripts in the previous table are run by the `commonDBUtility.ant` file from each component script, such as `configRecovery > commonDBUtility > execute createTable_Recovery.sql`. When the value `delayConfig=true` is in the response file, the SQL files are created, but they are not run. In this case, you must run the SQL manually after the configuration.

In the WebSphere Enterprise Bus Logger Mediation component, you can configure each message logger primitive to use a different data source and a different database.

Exported scripts

Scripts are created for any option that you selected on the Profile Management Tool panel to configure the common database. The scripts contain only basic creation statements for databases, tables, and indexes. The database administrator must use database native commands to run these scripts.

The names of the scripts are `configCommonDB.bat` for Windows, and `configCommonDB.sh` for UNIX-based operating systems.

Database scripts are exported to the

profile_root/dbscripts/CommonDB/dbType/dbName directory.

Planning to configure the Common Event Infrastructure database

The Common Event Infrastructure (CEI) database specifications list the types of supported databases, script locations, profile configuration types, and necessary user ID privileges. You can optionally use the CEI database to store events that are captured when it is monitoring IBM Business Process Manager.

The CEI database is an internal device and you do not interact directly with it. You must use the supported CEI programming interfaces for all interactions with the CEI database.

When you run the Profile Management Tool, you do not automatically create the CEI database. If you want to store CEI events, you must create the database manually for a stand-alone profile and for each instance of a CEI server in a network deployment environment.

Planning to configure the Business Process Choreographer database

Business Process Choreographer requires a database. For a non-production configuration, the Business Process Choreographer database can reside in the common database. For a production configuration, the Business Process Choreographer database should be a separate, high-performance database system.

Business Process Choreographer database requires one database, however some of the optional functions also require a database. The complete planning necessary for Business Process Choreographer is described in [Planning to configure Business Process Choreographer](#).

The following list provides links to the Business Process Choreographer topics that are related to planning databases.

- For information about the choices and restrictions for the Business Process Choreographer database, refer to [Planning the topology, setup, and configuration path](#).
- If you are not planning a simple sample Business Process Choreographer configuration, perform [Planning the databases for Business Process Choreographer](#).
- To plan the user IDs required to create and connect to the Business Process Choreographer databases, complete the planning tables related to databases in [Planning security, user IDs, and authorizations](#).

Planning to configure the messaging engine database

The messaging engine database specifications list supported database type, scripts and their locations, profile creation types, and necessary user ID privileges.

The messaging engine database is used to store operating information. Essential objects that the messaging engine needs for recovery in the event of a failure are also stored.

The messaging engine database is used by the message engines for Service Component Architecture (SCA) modules, Business Process Choreographer, and Common Event Infrastructure (CEI). The default database name for the SCA messaging engine is SCADB. For the other messaging engines, the default database name is MEDB. The default schema name is IBMWSSIB.

Important: Multiple schemas are not supported by all database types. For more information, see your database documentation.

In a stand-alone environment, you can use the administrative console to configure your SCA messaging engine. In a patterned network deployment environment, the messaging engines are configured during deployment environment creation. For a custom network deployment environment, you need to configure the messaging engines manually.

You have control over the messaging engine databases. For example, you can create a database for each messaging engine or you can use a single database for all the messaging engines. Each messaging engine must have either its own database or a schema.

Supported database types

The messaging engine database can use the following database products:

Table 45. Supported database products

Database Types	Considerations
DB2 Express	Used as the default database type for a stand-alone profile.
DB2 Universal	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 Data Server	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 for z/OS v8 DB2 for z/OS v9	Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
Microsoft SQL Server (Microsoft)	
Oracle	You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas.

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see Database privileges and User ID or schema name privileges.

For a network deployment environment, you need all necessary permissions for user privileges specified during configuration from the administrative console.

Important: For DB2 V9.7, grant the appropriate authority to the newly created user, because the user creation process does not automatically grant the user the necessary authority.

Database Management Service (DBMS) instances

Each messaging engine has its own database or schema:

- One is used to host each messaging engine for the Service Component Architecture system bus.
- Another is used to host each messaging engine for the Service Component Architecture application bus.
- Another is used to host each messaging engine for the Common Event Infrastructure bus.
- Another is used to host each messaging engine for the Business Process Choreographer bus.

The following list contains the naming conventions for the JDBC data source that the messaging engine uses to interact with the database:

- System bus: `<node><server>|<cluster>-SCA.SYSTEM.<cell>.Bus`
- Application bus: `<node><server>|<cluster>-SCA.APPLICATION.<cell>.Bus`
- CEI bus: `<node><server>|<cluster>-CEI.cellName.BUS`
- BPC bus: `<node><server>.-BPC.<cell>.Bus`

Configuration actions during profile creation

Network deployment

No messaging engine databases are created automatically. After the profile is created, you can use the Configure your Network Deployment Environment guided activity to configure a server or a cluster for SCA. To access this guided activity from the administrative console of the deployment manager, expand **Guided Activities** and click **Configure your Network Deployment Environment**.

You can view the SCA configuration of your server from the **Application servers** > *servername* > **Service Component Architecture** panel of the administrative console.

The following administrative tasks are performed during profile creation:

- Remote Destination Location:
 - `configSCAAsyncForServer`, `configSCAJMSForServer` (`remoteMELocation` is true)
 - `configSCAAsyncForCluster`, `configSCAJMSForCluster` (`remoteMELocation` is true)
- Local Destination Location:
 - `configSCAAsyncForServer`, `configSCAJMSForServer`
 - `configSCAAsyncForCluster`, `configSCAJMSForCluster`

For more information about these tasks, see “`configSCAAsyncForCluster` command” and “`configSCAAsyncForServer` command”.

When you perform asynchronous SCA configuration for a server or cluster, a messaging engine is created for the SCA system bus. When you perform the JMS element of the SCA configuration for a server or cluster, a messaging engine is created for the SCA application bus. For both messaging engines, you must create a database or schema.

The Business Process Choreographer messaging engines are created during Business Process Choreographer configuration. Business Process Choreographer is configured for patterned deployment environments only. For more information, see Planning the topology, setup, and configuration path.

To configure the Common Event Infrastructure messaging engine, use the `deployEventService` administrative task to configure the event server and the Common Event Infrastructure bus.

SQL scripts

No SQL scripts are created as part of the product. You can use existing base WebSphere Application Server scripts to create database and tables if necessary. To create the MEDB manually before it is

configured, use the **Servers > Server Types > WebSphere application servers > servername > Service Component Architecture** panel of the administrative console.

JDBC provider

Service Component Architecture

The JDBC provider is reused when the JDBC provider implementation class has to match with the one chosen in the advanced configuration. If the same database types are used, then the implementation classes usually match. If no matching JDBC provider is found in the `resource.xml` file, then the `jdbcc-resource-provider-templates.xml` file in the `templates/system` directory (profiles configuration) is searched for a matching JDBC provider. The provider is matched also against the implementation class.

Business Process Choreographer

The Business Process Choreographer reuses the JDBC provider for the SCA messaging engine. For more information, see [Configuring Business Process Choreographer](#).

Common Event Infrastructure

The JDBC provider creation for messaging engine database is similar to the approach followed in the creation of the CEIDB database.

Data source names

- System bus: : `_(node.server|cluster)-SCA.SYSTEM.cell.Bus/cel/cluster/server/node`
- Application bus: `_(node.server|cluster)-SCA.APPLICATION.cell.Bus/cell/cluster/server/node`
- Common Event Infrastructure: `_(node.server| cluster)-CEI.cellName.BUS/cluster/server/node`
- Business Process Choreographer: `_(node.server|cluster)-BPC.cell.Bus/cell/cluster/server/node`

Data source JNDI names

- System bus: `jdbcc/com.ibm.ws.sib/(node.server|cluster)-SCA.SYSTEM.cell.Bus/cell/cluster/server/node`
- Application bus: `jdbcc/com.ibm.ws.sib/(node.server|cluster)-SCA.APPLICATION.cell.Bus/cell/cluster/server/node`
- Common Event Infrastructure: `Jdbcc/ com.ibm.ws.sib /(node.server|cluster)-CEI.cellName.BUS/cluster/server/node`
- Business Process Choreographer bus: `jdbcc/com.ibm.ws.sib/(node.server|cluster)-BPC.cell.Bus/cell/cluster/server/node`

Restrictions

There are no known restrictions.

Tables

For information on the tables, see the topic “Data stores” in the [WebSphere Application Server Network Deployment information center](#).

Exported scripts

You can use the `sibDDLGenerator` script in `WAS_INSTALL_ROOT/bin` to create the SQL scripts for messaging engines database. Use the `sibDDLGenerator` script for creating SQL scripts for use in production environment especially on the DB2 for z/OS platform. For more information, see the “The `sibDDLGenerator` command”.

These scripts contain only basic create **database/tablespace/table** statements. A database administrator might still need to tailor these scripts to meet their database needs, especially on DB2 for z/OS.

Configuring messaging engine and server behavior when a data store connection is lost

The connection between a running messaging engine and its data store might sometimes be lost due to a failure or because you stop the database. You can determine the behavior of your system through a custom property on the messaging engine.

If you set the `sib.msgstore.jdbcFailoverOnDBConnectionLoss` custom property, you can enhance the automatic recovery of a highly available IBM Business Process Manager environment.

For information on the `sib.msgstore.jdbcFailoverOnDBConnectionLoss` property, including information how to set this property, see *Configuring messaging engine and server behavior when a data store connection is lost*.

Planning to configure the selector and business rules group database

Use the selector and business rules group database specifications to find information about supported database types, scripts and their locations, profile creation configuration actions, restrictions, table names, and user ID privileges.

When you install an application containing selector or business rule artifacts, the server stores the artifacts in database tables. You can then dynamically update them without changing the application. The selector and business rules group components use a database to hold the selector and business rule artifacts that are created in Integration Designer and installed on the server. If you change a selector through the administrative console, or a business rule through the Business Process Rules Manager, the database is updated with the latest information. The original artifacts in the enterprise archive (EAR) are not synchronized with updates that are made after the application is installed.

Supported database types

The selector and business rules group database can use the following database products:

Table 46. Supported database products

Database Types	Considerations
DB2 Express	Used as the default database type for a stand-alone profile.
DB2 Universal	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 Data Server	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 for z/OS v8 DB2 for z/OS v9	Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
Microsoft SQL Server (Microsoft)	
Oracle	You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas.

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see Database privileges and User ID or schema name privileges.

Database Management Service (DBMS) instances

Only one database instance is available in a stand-alone configuration or network deployment configuration (server or cluster of servers). All selectors and business rules for the server in a stand-alone configuration or in a cell use the same repository. There is no support for using separate database instances for different selectors or business rules.

Configuration actions during profile creation

Stand-alone profile

During profile creation, the Common database uses **createTable** to create selector and business rules tables. If you choose the default configuration, the default database is DB2. However, you cannot use multiple servers because DB2 allows only one Java virtual machine to access the database at a time.

Deployment manager profile

For network deployment configuration, the common database for the cell is set up with the appropriate tables for selector and business rule group components. All selectors and business rules for the server in a deployment manager or custom profile use this database.

SQL scripts

For more information about the location of SQL scripts, refer to Planning to configure the common database.

JDBC provider

The selector and business rules group database uses the common database data source and JDBC provider:

Data source name:

- **WPS DataSource**

Data source JNDI name:

- **jdbc/WPSDB**

Restrictions

If you uninstall applications that contain selectors or business rules from the server or cell, artifacts stored in the common database or RepositoryDB databases are not removed. For information on how to manually remove the artifacts, see Removing business rule and selector data from the repository.

Tables

For selectors and business rule groups, there are three tables that you can use to hold the artifacts:

- AppTimestamp
- ByteStore
- ByteStoreOverflow

Exported scripts

The scripts are exported to the same location as the scripts for the common database. For more information about the location of exported scripts, refer to Planning to configure the common database.

Planning to configure Business Process Choreographer

Plan your Business Process Choreographer setup, configuration parameters, and Business Process Archive.

1. Perform Planning the topology, setup, and configuration path.
2. Depending on your chosen configuration path, perform one of the following:
 - For “Basic sample”, perform Planning to create a basic sample Business Process Choreographer configuration.
 - For “Sample with organization”, perform Planning to create a sample Business Process Choreographer configuration including a sample organization.
 - For “Production deployment environment”, perform Planning to use the administrative console's deployment environment wizard.
 - For “Flexible custom configuration”, perform Planning for a custom Business Process Choreographer configuration.
 - For “Business Process Archive”, perform Planning for the Business Process Archive.

You have planned everything you need to be able to perform tasks for Configuring Business Process Choreographer.

Business Process Choreographer overview

Describes the facilities provided by the Business Flow Manager, Human Task Manager, and Business Process Archive Manager.

Business Process Choreographer is an enterprise workflow engine that supports both BPEL processes and human tasks in a WebSphere Application Server environment. These constructs can be used to orchestrate services as well as integrate activities that involve people in business processes. Business Process Choreographer manages the lifecycle of BPEL processes and human tasks, navigates through the associated model, and invokes the appropriate services.

Business Process Choreographer provides the following facilities:

- Support for BPEL processes and human tasks. Allows you to model your business process using the Web Services Business Process Execution Language (WS-BPEL, abbreviated to BPEL). With human tasks, you can use the Task Execution Language (TEL) to model activities that involve people. Both BPEL processes and human tasks are exposed as services in a service-oriented architecture (SOA) or Service Component Architecture (SCA); they also support simple data objects and business objects.
- Application programming interfaces for developing customized applications for interacting with BPEL processes and human tasks.
- Human workflow widgets as part of Business Space. These widgets allow you to manage work, create tasks for other people, and initiate services and processes.
- Business Process Choreographer Explorer. This web application allows you to administer BPEL processes and human tasks.
- Business Process Archive Manager. When configured, this provides a separate database, to which completed process instances and human tasks can be moved by running an administrative script. This can help to maintain the performance of your Business Process Choreographer database. An API is also available so that you can create your own client that can work with archived instances.

The Business Process Archive Explorer is a web application that allows users to browse or delete instances that have been moved to the archive database. For more information, see “BPEL process archiving overview.”

- By default, new Business Process Choreographer configurations benefit from the performance improvements resulting from using shared work items.
- To monitor and report on BPEL processes, use IBM Business Monitor.

Business Process Choreographer Explorer overview

Business Process Choreographer Explorer is a web application that implements a generic web user interface for interacting with BPEL processes and human tasks.

You can configure one or more Business Process Choreographer Explorer instances on a server or cluster. To connect a client such as the Business Process Choreographer Explorer to a Business Process Choreographer configuration, it is sufficient to have a Process Server installation with a WebSphere profile, or a Process Server client installation – it is not necessary to have Business Process Choreographer configured on the server or cluster. The Process Server client installation is the only infrastructure that you need.

The client installation does not include the Business Process Choreographer Explorer. You can use the deployment manager, administrative console, or an administrative script to install the Business Process Choreographer Explorer on the servers in the Process Server client installation as well.

A single Business Process Choreographer Explorer can only connect to one Business Process Choreographer configuration, though it does not have to connect to a local configuration. However, you can configure multiple instances of the Business Process Choreographer Explorer on the same server or cluster, and each instance can connect to a different Business Process Choreographer configuration.

When you start Business Process Choreographer Explorer, the objects that you see in the user interface and the actions that you can take depend on the user group that you belong to and the authorization granted to that group. For example, if you are a business process administrator, you are responsible for the smooth operation of deployed BPEL processes. You can view information about process and task templates, process instances, task instances, and their associated objects. You can also act on these objects; for example, you can start new process instances, create and start tasks, repair and restart failed activities, manage work items, and delete completed process instances and task instances. However, if you are a user, you can view and act on only those tasks that have been assigned to you.

You cannot use the Business Process Choreographer Explorer to view instances that have been moved to a Business Process Archive database. That requires a Business Process Archive Explorer instance that connects to the Business Process Archive Manager configuration. For more information about archiving see the BPEL process archiving overview.

BPEL process archiving overview

If you configure the Business Process Archive Manager, you use a script to move completed process instances and human tasks from the Business Process Choreographer database to an archive database.

By regularly performing archiving, you can prevent the runtime database from filling up with old objects, which over time, can degrade the database performance. You can use a Business Process Archive Explorer or the Business Process Archive Manager API to access processes and tasks that were moved to the archive database. Because it is not possible to move data from an archive database back to a runtime database, using this archiving facility does not provide any backup protection.

Architecture

The business process archiving facility consists of the following elements:

- Business Process Archive Manager

- Business Process Archive database
- The archive.py script
- Business Process Archive Explorer
- Business Process Archive Manager EJB API

Business Process Archive Manager

The Business Process Archive Manager must be configured before you can use it. The following conditions apply:

- Business Process Archive Manager must be configured in a network deployment environment and requires Service Component Architecture (SCA) and messaging services.
- A Business Process Choreographer configuration can only use Business Process Archive Manager configurations that are in the same cell.
- A Business Process Archive Manager configuration can only be used to archive data from one Business Process Choreographer configuration.
- Each Business Process Archive Manager configuration must have its own Business Process Archive database.
- A deployment target can only have one configuration of either Business Process Archive Manager or Business Process Choreographer.
- Applications that contain BPEL processes or human tasks cannot be deployed to a deployment target that is configured with the Business Process Archive Manager.
- For each Business Process Choreographer configuration, you can have one or more Business Process Archive Manager configurations.

Minimum configuration

The following figure illustrates using one Business Process Archive Manager configuration for a Business Process Choreographer configuration. This setup is sufficient to prevent database performance problems caused by the runtime database filling up with completed instances. Both configurations must be in the same cell.

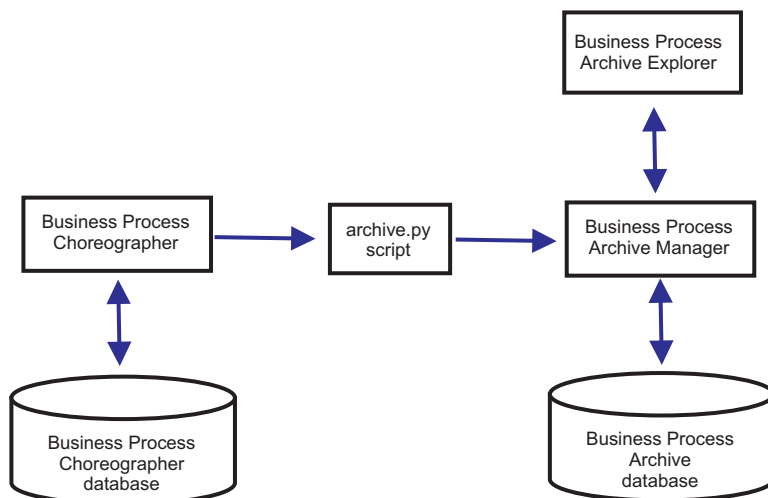


Figure 8. Business Process Choreographer using one Business Process Archive

Multiple configurations

It is also possible to use multiple Business Process Archives to archive the instances from a Business Process Choreographer configuration.

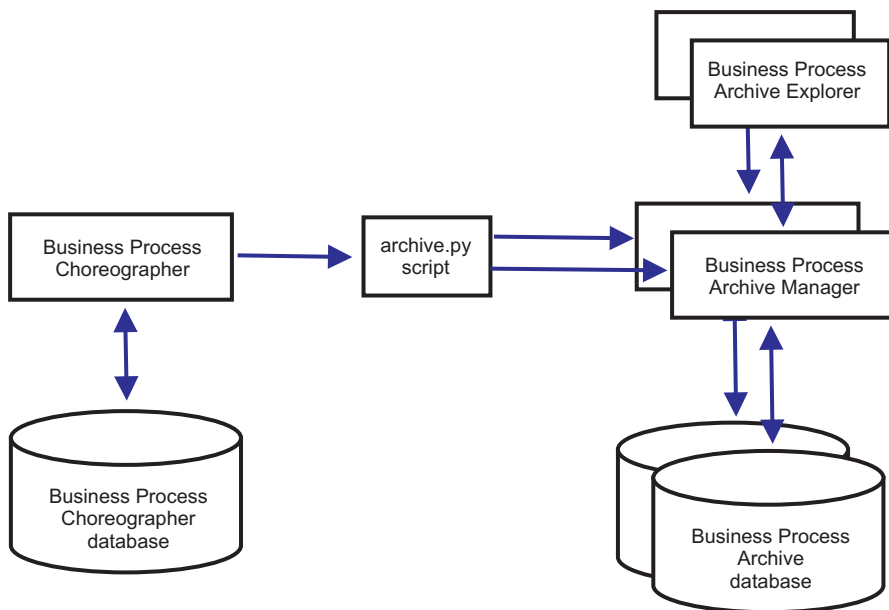


Figure 9. Business Process Choreographer using two Business Process Archives

For example, if different departments work with different BPEL processes, it might make sense to have an archive per department, and move completed instances of each departments' business processes to their own archive, so that the members of one department cannot access the archived process instances of another department.

Note: Using more than one Business Process Archive will not improve the performance of archiving operations.

Business Process Archive database

Each Business Process Archive Manager requires its own database. The database must be of the same type and structure as is used for the Business Process Choreographer database. The default name for the archive database is BPARCDB.

The archive.py administrative script

A WebSphere system administrator can run this script to archive data from the runtime database of one Business Process Choreographer configuration to the archive database of a Business Process Archive Manager configuration. Various parameters can be specified to control which instances will be archived, how many to archive in total, and how many to archive in each database transaction. The source and destination are specified by their deployment servers or clusters. For more information about this script, see Archiving completed BPEL process and task instances.

Restriction: The following restrictions apply:

- You cannot transfer objects from an archive database back to a Business Process Choreographer database, nor to another archive.
- The first time that you archive instances to a new archive database, the identity of the Business Process Choreographer configuration is written to the database, and in future, only instances from that configuration can be archived to that archive database.
- When instances are successfully moved to the archive, they are deleted from the Business Process Choreographer database, which generates a deletion event for the common event infrastructure (CEI) and the audit log. But it is not possible to identify that the deletion event was caused by an archiving action rather than by some other delete action, for example, cleanup service, user initiated delete action, delete script, or automatic deletion after successful completion.

- You cannot archive to different archives at the same time. Parallel invocations of the `archive.py` script are serialized.
- You cannot archive a process instance that has the same process name as any other process instance in the archiving database.
- You cannot archive a process instance that has the same values for its correlation set as another process instance in the archiving database.
- If you archive instances of a process template, then undeploy and redeploy the identical process template with the valid from date unchanged, you cannot archive any new instances of that process template to the same archive database. This is not an issue for normal process template versioning, where a different valid from date is used.

However, even if one of the restriction prevents you from archiving certain process instances to one archive database, you can archive those process instances to a different archive database, for which the restriction conditions are not true.

Business Process Archive Manager EJB API support

Only a subset of the actions that are available using the Business Flow Manager and Human Task Manager EJB APIs can also be used against a Business Process Archive Manager configuration to perform read and delete operations on process instances and human tasks that are in the associated archive database. The other APIs are not supported by the Business Process Archive Manager.

A new method, **OperationMode** `getOperationMode()` is provided, which indicates whether the client is connected to a Business Process Choreographer configuration or a Business Process Archive Manager configuration. This can be used to write custom clients that can connect to and operate appropriately on runtime configurations and archive configurations.

For more information about the Business Process Archive Manager API refer to the JavaDoc for the packages `com.ibm.bpe.api` and `com.ibm.task.api`.

Business Process Archive Explorer

The Business Process Archive Explorer is very similar to the Business Process Choreographer Explorer except that it connects to an archive database associated with a Business Process Archive Manager configuration. Like the Process Choreographer Explorer, the Business Process Archive Explorer must be configured before it can be used.

There are two ways to configure a Business Process Archive Explorer:

- If you run the `bpeconfig.jacl` script to create a Business Process Archive Manager configuration, there is an option to also configure a Business Process Archive Explorer instance on the same deployment target.
- If you only want to configure a Business Process Archive Explorer instance, you can run the `clientconfig.jacl` script interactively, or in batch mode using the **-operationMode ARCHIVE** option.

Depending on your authorization, you can use the Business Process Archive Explorer to browse instances and possibly delete instances too. You cannot update instances or create new instances.

Authorization

The actions that can be performed using the Business Process Archive Manager EJB API or the Business Process Archive Explorer depends on the following Java Platform, Enterprise Edition (Java EE) roles:

- Users who are in the Business Process Archive Manager system monitor role can read and view all process instances and all task instances in the archive database.
- Users who are in the Business Process Archive Manager system administrator role can also delete any top-level process instances and top-level task instances in the archive database.
- Users who are neither in the system monitor nor the system administrator roles can only see instances that they created or started themselves, but they cannot view any details about the instances.

- No one (not even users in the system administrator roles) can modify any of the data associated with any instances in the archive database.
- Instance-based authorization information, such as the potential owner or reader information, is not archived. Therefore, this data is not available in the archive. The only exception to this is the information about the starter and creator of processes and tasks.
- Users must be in the **WebClientUser** role to use the Business Process Archive Explorer.

Which data is archived

Only top-level process instances and top-level standalone human task instances that have reached one of the end states (**Finished**, **Terminated**, **Failed**, or **Expired**) can be moved to the archive database. When a top-level instance is archived, certain data is also moved with it to the archive, and other data is deleted.

For completed top-level process instances, including business state machine instances:

- Instance data such as activities, variables, inline human tasks, input messages, and output messages are moved.
- Child processes and related data are moved recursively.
- If any related metadata such as process templates and task templates are not already in the archive database, a copy of them is created.
- Query tables and stored queries are neither moved nor copied to the archive database.
- Work items associated with an archived instance are deleted without being archived.

For completed top-level standalone human tasks:

- Instance data such as input messages and output messages are moved.
- Escalation instances are moved.
- Child tasks including follow-on tasks are moved.
- If related metadata such as task templates are not already in the archive database, a copy of them is created.
- Work items associated with an archived instance are deleted without being archived.

Metadata

Additional metadata, such as process and task template information, is copied to the archive when necessary, to allow the archived data to be interpreted and displayed correctly. The metadata in the archive database is deleted when it is no longer required, that is, when the last process instance or human task that references the metadata is deleted.

What is not archived

Other Business Process Choreographer data, such as configuration data, XSD and WSDL artefacts, SCA modules, applications, work baskets, business categories, business rules, messages, and audit trail data, cannot be moved to the archive.

Overview of configuring a business process archive

If you want to configure a business process archive, you must plan to perform the following steps:

1. Perform “Planning for the Business Process Archive” on page 122.
2. Ensure that all the prerequisites are met:
 - The Business Process Choreographer configuration that the archive will be connected to must already be working.
 - The deployment target for the business process archive must already exist, and not have a Business Process Choreographer or Business Process Archive Manager configuration on it.

Important: The Service Component Architecture (SCA) on this deployment target must be configured to use the identical values for **Bus Member Location**, **Database Instance**, and **Schema**

as the SCA configuration on the deployment target of the associated Business Process Choreographer configuration. This allows it to share the existing messaging infrastructure.

3. Run the `bpeconfig.jacl` script.

You can only create a Business Process Archive Manager configuration using the `bpeconfig.jacl` script. The script can also create the first Business Process Archive Explorer configuration on the same deployment target. If you want further Business Process Archive Explorer configurations (on the same or different deployment target) you must use the `clientconfig.jacl` script.

4. If the Business Process Archive database does not already exist, it must be created before the Business Process Archive Manager is started.

5. Restart the server or cluster where the Business Process Archive Manager is configured.

6. Verify that the archiving facility works:

- If you have configured the Business Process Choreographer Explorer, use it to identify some BPEL process instances or human task instances that are in an end state that you want to move to the archive database.
- Run the `archive.py` script to move some completed BPEL process instances or human task instances to the archive.
- Direct a browser to the URL for the Business Process Archive Explorer to verify that the instances are in the archive.

Shared work items

Using shared work items is an optimization to improve the performance of database queries and reduce the database input/output. If you create a new Business Process Choreographer configuration, it automatically uses shared work items. If you migrated your Business Process Choreographer configuration using shared work items is optional.

What are shared work items?

Work items are created in the Business Process Choreographer database for each combination of human task instance and all people who are permitted to perform actions on that instance. If the number of human task instances and the number of users increases significantly, the number of work items in the database can grow so large that the database performance is affected.

The idea behind shared work items is an optimized implementation of work items, such that redundant information about work items is not stored in the database. This can improve the performance of many BPEL process and human task queries. Whether shared work items are being used, or not, is transparent to the users of Business Process Choreographer APIs.

How to decide whether to add support for shared work items

It is only if you migrate your Business Process Choreographer configuration from Version 7.0.0.2 or earlier, that the support for shared work items is not activated. Although activating is optional, certain factors indicate that using shared work items will improve performance.

- If you only have hundreds or a few thousands of human task instances in your system, there is probably little or no performance advantage in migrating to using shared work items.
- If you have hundreds of thousands of human task instances in your system, and your task list and process list queries are taking too long, migrating to using shared work items should improve the performance.
- Using shared work items can improve the performance of the following queries:
 - `query()`
 - `queryAll()`
 - `queryEntities()`
 - `queryEntityCount()`

- queryRows()
- queryRowCount()

Restrictions

Some queries can only be resolved using the non-shared work items. This means that enabling support for shared work items will not improve the performance of the following types of queries:

- Queries that run against the WORK_ITEM view exclusively.
- Queries that reference one or more of the following columns in the WORK_ITEM view:
 - ASSOC_OBJECT_TYPE
 - ASSOC_OID
 - OBJECT_ID
 - CREATION_TIME
 - OBJECT_TYPE
- Queries referencing a custom table that is directly joined with the WORK_ITEM view.
- Queries that were created as materialized views.
- Queries that use authorization with inherited work items.

Planning the topology, setup, and configuration path

Your choice of topology and setup affects which Business Process Choreographer configuration paths you can use.

The different configuration paths vary in complexity, flexibility, and their support for different topologies and databases.

1. Be aware that you must choose between several different configuration paths for creating a Business Process Choreographer configuration.
 - The “Basic sample” only includes the following:
 - Business Process Choreographer runtime configuration.
 - Business Process Choreographer Explorer.
 - The “Sample with organization” also includes the following:
 - A people directory that is preconfigured with 15 users in a sample organization.
 - Substitution and group work items are enabled.
 - “Production deployment environment”
 - “Flexible custom configuration”

For most configuration paths, you have a choice of configuration tools.

Note: An additional “Business Process Archive” configuration path only configures a Business Process Archive Manager.

2. Be aware of the different configuration tools that you can use to configure Business Process Choreographer.

Installer

The Installer provides an easy way to create a non-production system.

- The combination of a typical install and using DB2 for the common database is one way to get the “Basic sample” Business Process Choreographer configuration.
- The combination of a custom install and using DB2 Express for the common database is the only way to get the “Sample with organization” Business Process Choreographer configuration

Profile Management Tool

Profile Management Tool provides two ways to create a non-production system with a “Basic sample” Business Process Choreographer configuration.

- If you use DB2 for the common database, creating a typical profile for a Process Server or a Process Center will include a “Basic sample” Business Process Choreographer configuration.
- Creating an advanced profile and selecting the Business Process Choreographer option includes the “Basic sample” Business Process Choreographer configuration, which can use any supported database.

Administrative console's deployment environment wizard

Can be used to create a “Production deployment environment” Business Process Choreographer configuration, based on a deployment environment pattern.

Administrative console's Business Process Choreographer configuration page

You can use this administrative console page to configure a “Flexible custom configuration” Business Process Choreographer production system on a server or cluster. It provides the opportunity to set many configuration parameters, which need detailed planning. This page does not configure the Business Process Choreographer Explorer, which you can configure using its own configuration page in the administrative console, or by running a script. This configuration path is most suitable for creating production systems.

bpeconfig.jacl configuration script

You can use this script to configure a “Flexible custom configuration” Business Process Choreographer production system and all the necessary resources on a given server or cluster. You can run the script interactively, or if you provide all the necessary parameters, it can run in batch mode for repeatable automation. It can create a local database, the necessary messaging resources, and can optionally configure the Business Process Choreographer Explorer. For some database systems, it can also create a remote database. This configuration path is most suitable for creating production systems.

You can also use this script to configure a Business Process Archive Manager, but not on the same deployment target where Business Process Choreographer is configured.

clientconfig.jacl configuration script

You can use this script to configure one of the following:

- A Business Process Choreographer Explorer.
- A Business Process Archive Explorer.

3. Be aware that some of the configuration paths have restrictions that limit their suitability for production systems: For example:
 - If you create a Business Process Choreographer configuration that uses the common CMNDB database, it will not be suitable for a high-performance system.
 - After experimenting with one of the sample configurations, if you want to use the same profile to create a new configuration that is suitable for a production system, you must remove the sample configuration first. If you want to keep your sample configuration, you must create one or more separate profiles for the production setup.
4. Identify the main criteria for deciding which configuration path to use. Use the following table to identify choices and constraints:

Table 47. Criteria for selecting a configuration path

Choices		Restrictions		Suitable configuration path	
Are you planning a production system?	What is the deployment target?	Type of Business Process Choreographer configuration	Can it use a separate database?	Which message stores are supported for the messaging engine?	Configuration path name and possible configuration tools
No	Stand-alone server		Yes, the BPEDB database can be any supported database.	Only the shared DB2 CMNDB database	" Basic sample " using the Profile Management Tool to create an advanced Process Server or Process Center profile, and select the Business Process Choreographer option.
		Basic sample (without the sample organization)	No, the BPEDB database is created (using the schema name BPEDB) in the common database, CMNDB, which can only be DB2.		" Basic sample " using the Profile Management Tool to create a typical Process Server or Process Center profile.
		Sample including a 15 person organization, and substitution is enabled.	No, the BPEDB database is created (using the schema name BPEDB) in the common database, CMNDB, which can only be DB2 Express.		" Basic sample " using the Installer to create a typical installation of Process Server or Process Center
					" Sample with organization " using the Installer to create a custom install of the Process Server.

Table 47. Criteria for selecting a configuration path (continued)

Choices			Restrictions		Suitable configuration path
Are you planning a production system?	What is the deployment target?	Type of Business Process Choreographer configuration	Can it use a separate database?	Which message stores are supported for the messaging engine?	Configuration path name and possible configuration tools
Yes	Cluster	Choice of deployment environment patterns: <ul style="list-style-type: none"> • Single Cluster • Remote Messaging • Remote Messaging, Remote Support, and Web • Remote Messaging and Remote Support 	Yes, the BPEDB database can be any supported database.	Any supported database.	“ Production deployment environment ” using the Administrative Console.
		Flexible custom configuration			“ Flexible custom configuration ” using one of: <ul style="list-style-type: none"> • bpeconfig.jacl script • Administrative console Business Process Choreographer configuration page
Yes	Any network deployment server or cluster where Business Process Choreographer is not configured.	Business Process Archive	The BPARCDB archive database must be separate from the BPEDB database. It must be of the same database type that is used by the Business Process Choreographer configuration.	Does not have its own messaging engine. Uses the same messaging engine as the Business Process Choreographer configuration.	“ Business Process Archive ” using: <ul style="list-style-type: none"> • bpeconfig.jacl script to configure a Business Process Archive Manager.

Note: It is also possible to use any of the configuration paths that are recommended for creating a production system to create a configuration that is not suitable for a production system. Consider the following options:

- a. Decide whether you are configuring a production system. Typically a production system requires high-performance, scalability, and security. For Business Process Choreographer, a production system should have its own BPEDB database.
- b. Decide whether the deployment target for the Business Process Choreographer will be a stand-alone server or a cluster.

- c. If you do not want to create a production system, decide whether a sample configuration on a stand-alone server will meet your needs. If so, decide whether you want the sample to include a sample people directory (populated with a sample organization) for people assignment and substitution enabled.

Note: The sample people directory uses the default file registry configured for the federated repositories, and includes all sample people with the same password “wid”. The WebSphere administration user ID is also added to the directory, using the password that was specified during profile creation. After the sample configuration has been created, you can use the administrative console to view which users and groups are available by clicking **Users and Groups**, then either **Manage Users** or **Manage Groups**.

- d. If you want to configure Business Process Choreographer on a cluster, depending on your performance requirements, decide whether the messaging engines and supporting applications, (such as the Business Process Choreographer Explorer and Common Event Infrastructure) will have their own cluster, or share one. The standard deployment environment patterns are:

Single cluster

The single cluster pattern is the simplest pattern. It defines one application deployment target cluster, which includes the messaging infrastructure and supporting applications.

Remote Messaging

The remote messaging pattern defines one cluster for application deployment and one remote cluster for the messaging infrastructure. The supporting applications are configured on the application deployment target cluster.

Remote Messaging, Remote Support, and Web

The remote messaging, remote support, and web pattern defines one cluster for application deployment; one remote cluster for the messaging infrastructure; one remote cluster for the supporting applications; and one cluster for Business Space and REST services related web applications.

Remote Messaging and Remote Support

The remote messaging and remote support pattern defines one cluster for application deployment, one remote cluster for the messaging infrastructure, and one remote cluster for the supporting applications.

- e. For a production system, plan to have a dedicated BPEDB database for Business Process Choreographer.
- f. Business Process Choreographer will use the same database as a message store that is used by SCA. Business Process Choreographer will use its own schema in the same database.
5. If you want to use the Business Process Archive Manager to move completed process instances and human tasks from the BPEDB database to an archive database, you must configure it on a deployment target where Business Process Choreographer is not configured. To perform the detailed planning now, perform “Planning for the Business Process Archive” on page 122.
6. If you want WebSphere Portal Server or a custom IBM Business Process Manager Advanced client to access Business Process Choreographer, perform “Planning for a remote client application” on page 126.
7. If you have application security enabled and you have a long-running process that calls a remote EJB method, Common Secure Interoperability Version 2 (CSIv2) identity assertion must be enabled when you configure CSIv2 inbound authentication.
8. If you will use human tasks, then WebSphere administrative security and application security must both be enabled.

You have planned the topology and know which configuration path and configuration tool you will use.

Related concepts:

“BPEL process archiving overview” on page 90

If you configure the Business Process Archive Manager, you use a script to move completed process

instances and human tasks from the Business Process Choreographer database to an archive database.

Related tasks:

“Planning for a remote client application” on page 126

Planning for a remote Business Process Choreographer client application that uses the Business Process Choreographer APIs and runs on an IBM BPM Advanced client installation.

Planning to create a basic sample Business Process Choreographer configuration

This basic sample configuration, for a stand-alone server, does not include a sample organization.

You have performed Planning the topology, setup, and configuration path, and have decided that you want the “Basic sample”, which is not suitable for a production system.

1. Be sure that you know which tool and which options you will use to get the “Basic sample” Business Process Choreographer configuration:

Using the Installer:

Performing a typical install of the Process Server or the Process Center will only configure “Basic sample” Business Process Choreographer configuration if you select to use a DB2 database.

Using the Profile Management Tool:

Typical profile creation

The “Basic sample” Business Process Choreographer configuration is only configured if you select to use a DB2 database.

Advanced profile creation

To get the “Basic sample” Business Process Choreographer configuration, you must select the Business Process Choreographer option. This configuration can use any supported database.

2. If you want the Human Task Manager to be able to send escalation e-mails, plan the following:
 - If there will not be a local Simple Mail Transfer Protocol (SMTP) mail server available, plan to change the mail session later to point to a suitable mail server.
 - Plan to change the sender address for the e-mails. Otherwise, it will use a dummy sender address.
3. Be aware that this sample configuration uses the WebSphere administrator user ID and password for the various Business Process Choreographer user IDs.

You have planned to create a basic sample Business Process Choreographer configuration.

Planning to create a sample Business Process Choreographer configuration including a sample organization

This sample includes a 15 person sample organization, which is suitable for experimenting with people assignment and substitution on a stand-alone server.

You have performed Planning the topology, setup, and configuration path, and have decided that you want the “Sample with organization” configuration.

This sample Business Process Choreographer configuration requires minimal planning.

1. Plan to use the Installer, and to perform a custom installation, using the **Stand-alone development IBM Business Process Manager Advanced - Process Server (qbpmaps)** option. The sample configuration uses DB2 Express.
2. If you want the Human Task Manager to be able to send escalation e-mails, plan the following:
 - If there will not be a local Simple Mail Transfer Protocol (SMTP) mail server available, plan to change the mail session later to point to a suitable mail server.

- Plan to change the sender address for the e-mails. Otherwise, it will use a dummy sender address.
3. Be aware that this sample configuration uses the WebSphere administrator user ID and password for the various Business Process Choreographer user IDs.

You have planned to create a sample Business Process Choreographer configuration including a sample organization.

Planning to use the administrative console's deployment environment wizard

For a production system, plan all the configuration parameters for Business Process Choreographer, including a separate database. For a non-production system, you can use a shared database.

You have performed Planning the topology, setup, and configuration path, and have selected the “Production deployment environment” configuration path.

When using the deployment environment wizard, you must select the deployment environment pattern, then you have the opportunity to change the default database parameters and authentication aliases for the WBI_BPC component, and enter other parameters for Business Process Choreographer.

1. If you do not have enough information or authority to create the whole configuration on your own, consult and plan with the people who are responsible for other parts of the system. For example:
 - You might need to request information about your organization's LDAP server, if it uses authentication you will need to request a user ID, and authorization.
 - If you are not authorized to create the database, your database administrator (DBA) must be included in planning the databases. Your DBA will need a copy of the database scripts to customize and run.
2. Perform “Planning security, user IDs, and authorizations” on page 106.
3. Decide which deployment environment pattern you will use:
 - Single Cluster
 - Remote Messaging
 - Remote Messaging, Remote Support, and Web
 - Remote Messaging and Remote Support
4. If you are planning to have dedicated databases for the following:
 - The BPEDB database for Business Process Choreographer, which can be changed in the wizard in a table row for the component WBI_BPC.
 - The BPEME database for the Business Process Choreographer messaging engine, which can be changed in the wizard in a table row for the component WBI_BPC_ME.

Plan the following parameters for each database, to enter on the wizard's database page:

Database name

The name of the database, for example, BPEDB or BPEME instead of the default value, CMNDB, which results in sharing the common database. The default value is only suitable for lower performance setups.

Schema

The schema qualifier to be used for each database.

Create Tables

If selected, the tables will be created automatically the first time that the database is accessed. For this option to work, the database must already exist, and the user name provided for creating the data source must have the authority to create tables and indexes in the database. If not selected, the tables will not be created automatically, and you must create the tables manually by running scripts. For a production system, clear this option, and plan to use the provided SQL scripts to set up the database.

User Name and Password

A user ID that has the authority to connect to the database and to modify the data. If the user ID has the authority to create tables and indexes in the database, then the option to create the tables automatically can be used, and when necessary, the database schema will be updated automatically after applying a service or fix pack.

Server The address of the database server. Specify either the host name or the IP address.

Provider

The JDBC provider.

Also plan the database-specific settings, which you can set by clicking **Edit** for the JDBC provider.

Table 48. Database-specific settings

Database / JDBC driver type	Database-specific settings
DB2 UDB – Universal driver	<ul style="list-style-type: none"> • User name • Password • Database name • Schema name • Server name • Server port number • Driver type • Description • Create tables
DB2 for z/OS	<ul style="list-style-type: none"> • Implementation type – Connection pool data source or XA data source • User name • Password • Database name • Schema name • Server name • Server port number • Storage group • Description
Microsoft SQL Server – Microsoft driver	<ul style="list-style-type: none"> • User name • Password • Database name • Server name • Server port number • Description • Create tables

Table 48. Database-specific settings (continued)

Database / JDBC driver type	Database-specific settings
Oracle – thin driver	<ul style="list-style-type: none"> • User name • Password • Database name • Schema name • Server name • Server port number • Driver type – thin • Description • Create tables

For more details about planning the databases, see “Planning the databases for Business Process Choreographer” on page 111.

5. Plan the user name for the Business Process Choreographer JMS authentication alias that you will enter during the Security step.
6. Plan the **Business Process Choreographer Explorer context root**, which defines part of the URL that browsers must use to reach the Business Process Choreographer Explorer.
7. Plan the security parameters for the Business Process Choreographer step. These user IDs and groups will be used for the Business Flow Manager and Human Task Manager:

Administrator User and Group

Plan a list of user IDs or a list or groups, or both, onto which the business administrator role is mapped.

Monitor User and Group

Plan a list of user IDs or a list or groups, or both, onto which the business monitor role is mapped.

JMS API Authentication User and Password

The run-as user ID for the Business Flow Manager message driven bean.

Escalation User Authentication User and Password

The run-as user ID for the Human Task Manager message driven bean.

Cleanup User Authentication User and Password

The run-as user ID for the Business Flow Manager and Human Task Manager cleanup service. This user must be in the business administrator role.

8. If you want to configure an email session for the Human Task Manager escalations, plan the following parameters for the Business Process Choreographer step:

Mail transport host

The host name or IP address where the Simple Mail Transfer Protocol (SMTP) email service is located.

Mail transport user and Mail transport password

If the mail server does not require authentication, you can leave these fields empty.

Business Process Choreographer Explorer URL

This URL is used to provide a link in generated emails so that a business administrator who receives an email notification can click the link to view the related business process or human task in their web browser.

9. If you are going to use the Business Process Choreographer Explorer, the Business Space, or a client that uses the Representational State Transfer (REST) API or the JAX web services API, decide on the context roots for the REST API and the JAX web services API.
 - The defaults for the Business Flow Manager are /rest/bpm/bfm and /BFMJAXWSAPI.

- The defaults for the Human Task Manager are /rest/bpm/htm and /HTMJAXWSAPI.
 - When configured on a server, or on a single cluster, or on multiple clusters that are mapped to different web servers, you can use the default values.
 - When configured in a network deployment environment on multiple deployment targets that are mapped to the same web server, do not use the default values. The context root for each Business Process Choreographer configuration must be unique for each combination of host name and port. You will have to set these values manually using the administrative console after configuring Business Process Choreographer.
10. If you want to use people assignment, perform “Planning for the people directory provider” on page 119.

You have planned to use the administrative console's deployment environment wizard.

Planning for a custom Business Process Choreographer configuration

Plan the configuration parameters and options for creating a custom configuration, using either the Administrative console's Business Process Choreographer configuration page or the bpeconfig.jacl configuration script.

You have performed Planning the topology, setup, and configuration path, and have selected the “Flexible custom configuration” configuration path.

1. Know which of the following you will use to configure Business Process Choreographer:
 - Administrative console's Business Process Choreographer configuration page
 - The bpeconfig.jacl configuration script
2. If you do not have enough information or authority to create the whole configuration on your own, consult and plan with the people who are responsible for other parts of the system. For example:
 - You might need to request information about your organization's LDAP server, if it uses authentication you will need to request a user ID, and authorization.
 - If you are not authorized to create the database, your database administrator (DBA) must be included in planning the databases. Your DBA will need a copy of the database scripts to customize and run.
3. “Planning security, user IDs, and authorizations” on page 106
4. “Planning the databases for Business Process Choreographer” on page 111
5. “Planning for the Business Flow Manager and Human Task Manager” on page 118
6. “Planning for the people directory provider” on page 119
7. “Planning for the Business Process Choreographer Explorer and Business Process Archive Explorer” on page 121
8. If you will use the Administrative console's Business Process Choreographer configuration page, make sure that you have planned all the values that you will enter on the configuration page.
9. If you will use the bpeconfig.jacl configuration script:
 - a. Make sure that you have planned all the options and parameter values that you must specify on the command-line, or in a batch file. The options and parameters are summarized in Using the bpeconfig.jacl script to configure Business Process Choreographer, and are described in detail in bpeconfig.jacl script file.
 - b. If you want, you can specify the parameter values in a response file.
 - c. If you will use a batch file to run the bpeconfig.jacl configuration script, create the batch file or shell script.

You have planned everything you need to be able to create a custom Business Process Choreographer configuration.

Perform topics/t2configovr.dita.

Planning security, user IDs, and authorizations

Plan the user IDs and authorizations for configuring Business Process Choreographer.

During configuration, you need to use various user IDs and you must specify other user IDs that will be used at run time. Make sure that you plan and create all user IDs before you start configuring Business Process Choreographer.

For a sample Business Process Choreographer configuration:

You only need the authority to create a new profile. In the Profile Management Tool, using the option to create a typical profile, when you enable administrative security, the Business Process Choreographer sample will also be configured. No other planning or user IDs are required, and you can skip this task.

For a high security configuration:

You must plan all user IDs in detail as described in this task.

For a low security configuration:

If you do not require full security, for example for a non-production system, you can reduce the number of user IDs that are used. You must plan all user IDs in detail, but you can use certain user IDs for multiple purposes. For example, the database user ID used to create the database schema can also be used as the data source user name to connect to the database at run time.

If you will use the `bpeconfig.jacl` script to configure Business Process Choreographer:

The user ID used to run the `bpeconfig.jacl` script must have the necessary rights for the configuration actions that the script will perform. Otherwise, you must specify user IDs as parameters for the script that have the necessary rights, in which case you must plan all user IDs in detail. For user IDs that can be specified as parameters to the `bpeconfig.jacl` script, the parameter names are included in the table. The profile must already exist. If WebSphere administrative security is enabled, you need a WebSphere administrator user ID in the configurator role that you can use to invoke the `wsadmin` tool.

If you will use human tasks:

- WebSphere administrative security and application security must both be enabled.
 - Do not enable the security option **Use realm-qualified user names**.
1. Print a hardcopy of this page so that you can write your planned values in the last column. Keep it for reference when you are configuring Business Process Choreographer, and keep it in your records for future reference.
 2. Plan the user ID you will use on the Process Server to configure Business Process Choreographer.

Table 49. Planning user IDs for Process Server

User ID or role	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
The user who configures Business Process Choreographer	Configuring	Logging onto the administrative console and running administrative scripts.	WebSphere administrator or configurator role, if WebSphere administrative security is enabled.	
		If you are going to run the <code>bpeconfig.jacl</code> script to configure the Business Process Choreographer.	When running the script, you must also provide any user IDs that are necessary for the options that you select. For more information see <code>bpeconfig.jacl</code> script file.	

3. Plan which people need access to subdirectories of `install_root`. If your security policy does not allow these people to be granted this access, they will need to be given copies of the files in the directories.

Table 50. Planning access to the subdirectories of *install_root*

User ID or role	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
Database administrator	Configuring	<p>Running the scripts to setup the following databases:</p> <p>BPEDB: This is the default name for the database for Business Process Choreographer.</p> <p>OBSRVDB: This is the default name for the database for the Business Process Choreographer Explorer reporting function.</p>	<p>If you use the bpeconfig.jacl script to configure Business Process Choreographer:</p> <p>Read access to (or a copy of) the <code>createSchema_BPC.sql</code> script for a DB2 for z/OS database or <code>createSchema.sql</code> for other databases, which bpeconfig.jacl generates in a subdirectory of the directory:</p> <ul style="list-style-type: none"> Linux UNIX <code>profile_root/dbscripts/ProcessChoreographer/</code> Windows <code>profile_root\dbscripts\ProcessChoreographer\</code> 	
			<p>If you want to review the database script files:</p> <p>Read access to (or a copy of the files in) the database scripts provided in the directory:</p> <ul style="list-style-type: none"> Linux UNIX <code>install_root/dbscripts/ProcessChoreographer/database_type</code> Windows <code>install_root\dbscripts\ProcessChoreographer\database_type</code> <p>Where <i>database_type</i> is one of the following:</p> <ul style="list-style-type: none"> DB2 DB2zOS Oracle SQLServer 	
Integration developer	Customizing	To use people assignment with a Lightweight Directory Access Protocol (LDAP) or Virtual Member Manager (VMM) people directory provider, you will have to customize a copy of the sample XSL transformation file.	<p>Either read access to the Staff directory, or a copy of the files in the directory:</p> <ul style="list-style-type: none"> Linux UNIX <code>install_root/ProcessChoreographer/Staff</code> Windows <code>install_root\ProcessChoreographer\Staff</code> <p>The integration developer will also need write access to a suitable directory to make the customized XSL transformation file available to the server.</p>	

- Plan the user IDs that will be used to create, configure, and access the database that is used by Business Process Choreographer.

Table 51. Planning user IDs for the BPEDB database

User ID or role	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
Database administrator	Before configuring	To create the BPEDB database.	Create the database.	

Table 51. Planning user IDs for the BPEDB database (continued)

User ID or role	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
Database administrator or an administrator who will run the bpeconfig.jacl script	Configuring	You or your database administrator must run Business Process Choreographer database scripts, unless you are using the default database.	For the BPEDB database: Alter tables, connect, insert tables, and create indexes, schemas, tables, table spaces, and views.	
Data source user name If you use the bpeconfig.jacl script, this is the -dbUser parameter.	Configuring	If you select the Create Tables option, this user ID is used to create the database tables.	To use the Create Tables configuration option, this user ID must also be authorized to perform the following actions on the BPEDB database: Alter tables, connect, insert tables, and create indexes, tables, and views.	
	Run time	The Business Flow Manager and Human Task Manager use this user ID to connect to the BPEDB database.	This user ID must be authorized to perform the following actions on the BPEDB database: Connect, delete tables, insert tables, select tables and views, and update tables.	
	After applying service or a fix pack	When necessary, the database schema is updated automatically after applying service. This only works if this user ID has the necessary database rights, otherwise schema updates must be performed manually.	This user ID must be authorized to perform the following actions on the BPEDB database: Alter, create, insert and select tables, connect to the database, create and drop indexes and views.	

- If you will have a separate database for the Business Process Choreographer's messaging engine message store, plan the user ID that will be used to access the database.

Table 52. Planning user ID for the preconfigured BPEME messaging engine database

User ID	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
Bus data source user name If you use the bpeconfig.jacl script, this is the -medbUser parameter.	Configuring and run time	This user name is used to connect to the BPEME database, and to create the necessary tables and index.	This user ID must be authorized to perform the following actions on the BPEME database: Connect, delete tables, insert tables, select tables and views, and update tables.	

- Plan the Business Process Choreographer user IDs for the Java Message Service (JMS).

Table 53. Planning user IDs for JMS

User ID	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
JMS authentication user	Run time	The authentication alias for the system integration bus. You must specify it when configuring Business Process Choreographer. If you use the bpeconfig.jacl script, this user IDs and its password are the parameters -mqUser and -mqPwd.	It must be a user name that exists in the WebSphere user registry. It is automatically added to the Bus Connector role for the Business Process Choreographer bus.	
JMS API authentication user	Run time	Any Business Flow Manager JMS API requests will be processed on using this user ID. If you use the bpeconfig.jacl script, this user IDs and its password are the parameters -jmsBFMRunAsUser and -jmsBFMRunAsPwd.	The user name must exist in the WebSphere user registry.	
Escalation authentication user	Run time	Any Human Task Manager escalations will be processed using this user ID. If you use the bpeconfig.jacl script, this user ID and its password are the parameters -jmsHTMRunAsUser and -jmsHTMRunAsPwd.	The user name must exist in the WebSphere user registry.	

- Plan which groups or user IDs, the Java EE roles for the Business Flow Manager and Human Task Manager will be mapped onto.

Table 54. Planning the security roles for the Business Flow Manager and Human Task Manager

User ID or role	When the user ID is used	What the user ID is used for	Planned list of user IDs, groups, or both
Administrator user	Run time	The system administrator and monitor security roles for both the Business Flow Manager and Human Task Manager are each mapped to a list of user IDs, groups, or both. The values defined here create the mapping that gives users in this role the access rights that they need. If you use the bpeconfig.jacl script, these users and groups correspond to the following parameters:	
Administrator group	Run time		
Monitor user	Run time		
Monitor group	Run time	<ul style="list-style-type: none"> • -adminUsers • -adminGroups • -monitorUsers • -monitorGroups 	

- Plan the user ID to use as the Java EE run-as role for administration jobs like the Business Flow Manager and Human Task Manager cleanup services and the process instance migration tool. This user ID must be a member of the administrator role user or group planned in Table 54.

Table 55. Planning the user ID for running administration jobs

User ID	When the user ID is used	What the user ID is used for	Planned user ID
Administration job user ID	Run time administration	This user ID is used to run administration jobs. If you use the bpeconfig.jacl script, this user ID and its password correspond to the -adminJobUser and -adminJobPwd parameters.	

- If you want human task escalations to send notification emails for specific business events, and your Simple Mail Transfer Protocol (SMTP) server requires authentication, decide which user ID will be used to connect to the email server.

Table 56. Planning the user ID for the email server

User ID or role	When the user ID is used	What the user ID is used for	Which rights the user ID must have	Planned user ID
Mail transport user	Run time	The Human Task Manager uses this user ID to authenticate against the configured mail server to send escalation emails. If you use the bpeconfig.jacl script, this is the -mailUser parameter. The password is the -mailPwd parameter.	Send emails.	

- If you will use people assignment for human tasks, and you will use a Lightweight Directory Access Protocol (LDAP) people directory provider that uses simple authentication, plan a Java Authentication and Authorization Service (JAAS) alias and an associated user ID that will be used to connect to the LDAP server. If the LDAP server uses anonymous authentication this alias and user ID are not required.

Table 57. Planning the alias and user ID for the LDAP server

User ID or role	When the user ID is used	What the alias and user ID are used for	Which rights the user ID must have	Planned alias and user ID
LDAP plug-in property: Authentication Alias	Run time	The alias is used to retrieve the user ID that is used to connect to the LDAP server. You specify this alias ID when customizing the properties for the LDAP plug-in, for example mycomputer/My LDAP Alias .	The JAAS alias must be associated with the LDAP user ID.	
LDAP user ID	Run time	This user ID is used to connect to the LDAP server.	If the LDAP server uses simple authentication, this user ID must be able to connect to the LDAP server. This user ID is either a short name or a distinguished name (DN). If the LDAP server requires a DN you cannot use the short name.	

- Create the user IDs that you have planned with the necessary authorizations. If you do not have the authority to create them all yourself, submit a request to the appropriate administrators, and enter the names of the user IDs that they create for you in this table.

You know which user IDs will be required when configuring Business Process Choreographer.

Planning the databases for Business Process Choreographer

Plan the databases for Business Process Choreographer.

Depending on your Business Process Choreographer setup, you might need to plan to create up to three databases, or none (if all the databases are put in the CMNDB common database). Each Business Process Archive Manager also needs a database, which can also reside in the common database. However, for production systems, you should plan to have all the Business Process Choreographer databases as separate, high-performance databases.

Business Process Choreographer can share a database with other process server components. The BPEDB database is used by the Business Flow Manager and the Human Task Manager. For a production system plan to have a dedicated database for each deployment target where Business Process Choreographer is configured.

If you have multiple Business Process Choreographer configurations, then each of these needs its own database or database schema. The Business Process Choreographer database tables cannot be shared between multiple Business Process Choreographer configurations.

The Business Process Choreographer messaging engines can either share the database used by the SCA messaging engines, or have their own BPEME database. For more information about which databases are supported for your selected configuration path, see Table 47 on page 98 or .

1. For a production system:

- a. If performance is important, plan to use a separate database for Business Process Choreographer, as described in “Planning the BPEDB database” on page 112, otherwise, plan to use the CMNDB common database.
- b. For high-load setups, for example, a large cluster with very high messaging rates, consider improving the performance by using a separate database for the Business Process Choreographer messaging engine. This allows the database logging to be parallelized, which can help to prevent it from becoming a bottleneck.
 - If you use the administrative console to configure Business Process Choreographer, and you want a separate database for the Business Process Choreographer messaging engine, perform “Planning the messaging engine database” on page 117, otherwise plan to use the default database that is used by the Service Component Architecture (SCA).
 - If you use the `bpeconfig.jacl` configuration script to configure Business Process Choreographer, Business Process Choreographer will use the same database as a message store that is used by SCA. Business Process Choreographer will use its own schema in the same database.
- c. Optional: Use the database design tool to interactively create the database design file and the SQL script files that the database administrator can use to create all three databases that you planned in the previous steps. There are significant advantages to using this tool:
 - You can run the tool as often as necessary to refine the database design parameters, without the risk of breaking them, rather than editing the provided template SQL files manually.
 - If you have used a database design file, the next time that you migrate to a newer version of this product, you can generate the schema update SQL scripts.
 - If you create a database design file for a test configuration, it is convenient to be able to make a copy of the design file and make minor changes to it for the databases for your production system.
 - Using the tool, you can also define the data sources for all three databases.

Important: When using the database design tool to create a deployment environment, after you have configured the common database, Business Process Choreographer is shown as being “complete”. This is because there is a valid default, that causes the tables for Business Process Choreographer to be created in the common database. However, this default is not suitable for

production systems. For a production system, make sure that you configure a dedicated database for each deployment target where Business Process Choreographer is configured.

2. For a non-production system where simplicity of setup is more important than performance, your options depend on the configuration path that you have chosen:
 - If you will use the Installer or Profile Management Tool to create the “Basic sample” or the “Sample with organization” Business Process Choreographer configuration, a separate default BPEDB database is created, which is also used by the Business Process Choreographer Explorer reporting function. For the Business Process Choreographer messaging engine, the default is to have a separate default database (BPEDB). If you use the Profile Management Tool, you can also select to share the CMNDB database.
 - If you will use the Profile Management Tool to create a deployment environment that includes a Business Process Choreographer configuration, Business Process Choreographer, and the Business Process Choreographer messaging engine will all use the CMNDB database. Therefore, you do not need to do any database planning for Business Process Choreographer.
3. If you will configure one or more Business Process Archive Manager configurations, perform “Planning the Business Process Archive database” on page 118 for each Business Process Archive Manager configuration.

You have planned all the databases for your Business Process Choreographer configuration.

Planning the BPEDB database:

Plan the database for Business Process Choreographer.

Business Process Choreographer requires a database. SQL scripts are provided for all supported database systems to create and administer the database schema. When a database is in place, JDBC access to the database has to be configured for Business Process Choreographer. Depending on the database system, your topology, the purpose of the setup, and the administrative tool you choose to use, some or all of the tasks to create the database and to configure JDBC access can be automated. For a production system, Business Process Choreographer should, in most cases, have its own database. However, if performance is not important, or if you are using DB2 for z/OS, you can also configure Business Process Choreographer to share a database with other IBM Business Process Manager components.

1. Make sure that your choice of BPEDB database and configuration path are compatible: The following databases are supported:
 - DB2 UDB for Linux, UNIX, and Windows
 - DB2 for z/OS
 - DB2 Express
 - Microsoft SQL Server
 - Oracle

If you have already decided how you are going to configure Business Process Choreographer, your choice of configuration path has implications on how you can create the database. If you have not yet decided which configuration path to use to configure Business Process Choreographer, identifying your database requirements will help eliminate the configuration paths that do not support your needs. For details about which databases are supported by each configuration path, see Table 47 on page 98 or .

2. If you do not need the performance, scalability, and security that is normally required for a production system, you can have the database objects created in a single table space on a database server that is local to IBM Business Process Manager. This minimizes the planning and effort necessary to create the database, but requires that the user ID used to access the database also has database administration rights. The options that you need to plan depend on the configuration path that you choose:
 - a. If you use the **Installer** or **Profile Management Tool** to get a sample Business Process Choreographer configuration, a separate default BPEDB database is created for Business Process Choreographer, which requires no further planning.

- b. If you use the administrative console **Deployment Environment wizard** to configure Business Process Choreographer, and it is sufficient for the default schema to be created in a single table space, plan to use a copy of the provided SQL script to create the BPEDB database.
- c. If you use the **bpeconfig.jacl** tool to configure Business Process Choreographer, identify which of the following conditions apply to you, based on the restrictions of the tool.
 - If you will run the bpeconfig.jacl script in interactive mode, you can select to create the tables in an existing database.
 - If you have a user ID with the authority to create the database objects, you can use the `-createDB yes` option, which causes the bpeconfig.jacl script to generate and run an SQL file to create the database objects in the default table space. In this case also plan to stop the server and use the `-conntype NONE` option for the **wsadmin** utility.
 - If you are using an Oracle database, the database must already exist. If you are using a DB2 for z/OS database, the database instance must already exist. For other database types, bpeconfig.jacl will try to create the database.
 - If any error occurs while creating the database or objects, you can use the generated SQL scripts as if you used the `-createDB no` option.
 - If you do not have a user ID with the authority to create the database objects, you must use the `-createDB no` option, which causes the bpeconfig.jacl script to generate an SQL file to create the database objects in the default table space, but it does not run the script. In this case, plan to ask your database administrator to customize and run the script for you.

For more information about the tool and other database parameters, see bpeconfig.jacl script file.

- d. If you use the administrative console's **Business Process Choreographer configuration page**:
 - To have the Business Process Choreographer database objects created in the common database CMNDB, plan to use the default database as the target for the Business Process Choreographer data source.
 - To reuse an existing database, plan to specify the existing database as the target for the Business Process Choreographer data source.
 - If you select the Create tables option, Business Process Choreographer will create the database objects that it needs in the default table space, the first time that it uses the database. This option cannot be used for a DB2 on z/OS database nor for a remote Oracle database. To use this option for a DB2 UDB database, the database must have **AUTOMATIC STORAGE YES** enabled.
 - To create the database using scripts, plan not to use the Create tables option.
 - e. Skip to step 12 on page 116.
3. Perform all of the following steps if you want a **high performance** database setup for Business Process Choreographer with the following characteristics:
 - The database is only used by Business Process Choreographer.
 - Ideally, the database is on a dedicated server that has plenty of memory to allow complex queries and read operations to be processed efficiently. The database can be local to the Process Server system. The database host requires a fast storage subsystem, for example, hardware-assisted disk arrays such as network-attached storage or a storage area network usually offer better performance than single disks.
 - You can customize the allocation of tables space to disks for better performance.
 - You can use a different user ID to access the database to the one used to administer it.
 4. If you have not already planned the user IDs for the database, perform Table 51 on page 107.
 5. Plan the allocation of disks and table spaces. For a production system, take into account the results of your experiences during development and system testing. The size of your database depends on many factors.
 - Processes that run as microflows use very little database space.
 - Each process template can require tens or hundreds of kilobytes.

- If the nature of your processes results in a frequent transaction log updates, it might be necessary to allocate the transaction logs to dedicated disks.

If you will use individual disks, and your database system supports allocating database tables to different disks, plan how many disks you will use and how you will allocate them.

For DB2 for z/OS, a table space is created for each table, and additional larger object (LOB) table spaces for LOB columns.

Plan the table spaces if you use one of the following:

- DB2
- Oracle

Your goal, when planning the table spaces, is to distribute input and output operations over as many disk drives as possible. By default, the scripts create the following table spaces:

AUDITLOG

Contains the audit trail tables for processes and tasks. Depending on the degree of auditing that is used, access to tables in this table space can be significant. If auditing is turned off, tables in this table space are not accessed.

BPETEMP8K

Contains the temporary table space to support the 8 KB buffer pools that are needed when migrating the database schema.

BPETS8K

Contains the table space to support the 8 KB buffer pools that are needed when migrating the database schema.

INSTANCE

Holds process instance and task tables. It is always used intensively, regardless of the kind of long-running process that is run. Where possible, locate this table space on its own disk to separate the traffic from the rest of the process database. The growth rate of this table space depends on your business applications.

LOBTS

Stores large data objects belonging to instances of BPEL processes and human tasks. It is used intensively and its growth rate correlates with the number of instances.

SCHEDTS

Contains the tables that are used by the scheduling component. Access to tables in the scheduler table space is usually low, because of the caching mechanisms used in the scheduler. The size of this table space correlates to the number of instances.

STAFFQRY

Contains the tables that are used to temporarily store staff query results that are obtained from staff registries like Lightweight Directory Access Protocol (LDAP). When BPEL processes contain many human activities, tables in this table space are frequently accessed.

TEMPLATE

Contains the tables that store the template information for processes and tasks. The tables are populated during the deployment of an application. At run time the access rate is low. The data is not updated, and only new data is inserted during deployment.

WORKITEM

Holds the tables that are required for work item processing. Work items are used for human task interaction. Depending on the number of human tasks in the BPEL processes, access to the tables in this table space can vary from a low access rate to significantly high access rate. The access rate is not zero, even when no explicit human tasks are used, because work items are also generated to support administration of long-running processes.

They can be all on one high-performance RAID-array, but each table space should be in a different file to allow parallel access. Keep in mind that for a given number of disks, using a RAID

configuration will have better performance than allocating table spaces to separate disks. For example, for a DB2 database that is running on a dedicated server with N processors, consider using the following guidelines:

- For the table spaces, use a RAID-1 array with 2*N primary disks, 2*N mirror disks, and a stripe size of 256 kb.
- For the database transaction log, use a RAID-1 array with 1.5*N primary disks, 1.5*N mirror disks, and a stripe size of 64 kb.

If you are using a DB2 database, running on a four-processor server, and will use 15 disk drives on a RAID controller, consider using the following allocations:

- One disk for the operating system and paging (known as the page file on Windows, paging space on AIX®, and swap space on Solaris).
- Use eight disks in a RAID-1 configuration (four primary disks and four mirrors) as one logical disk for the database control files and table spaces, using a stripe size of 256 kb.
- Use six disks in a RAID-1 configuration (three primary disks and three mirrors) as one logical disk for the database transaction log, using a stripe size of 64 kb.

If you are using an Oracle database, consider the following guidelines:

- Stripe and mirror everything (SAME) for all files, across all disks, using a one megabyte stripe width.
- Mirror data for high availability.
- Subset data by partition, not by disk.
- Use the Automatic Storage Management (ASM) file system.
- Create a partition (for the tables space) that is on the outside half of the disk drives.
- Make sure that you customize the distribution of the table spaces over the available disks when you create the BPEDB database schema.
- The redo log files are on dedicated disks.
- Enable automatic memory management.
- Set the statistics level to **TYPICAL**.
- Customize the maintenance window for automatic maintenance jobs.

If you are using a DB2 for Linux, UNIX, and Windows database, consider the following guidelines:

- Make sure that you customize the distribution of the table spaces over the available disks when you create the BPEDB database schema.
 - The transaction log files are on dedicated disks.
 - You might need to tune the lock timeout and deadlock check time configuration parameters to suit your setup.
 - Plan to edit the createTablespace.sql to change the table spaces from the default value **MANAGED BY AUTOMATIC STORAGE** to **MANAGED BY SYSTEM USING (storage location)**, which allows more control over disk usage and performance.
6. Plan that you or your database administrator will customize the SQL scripts that create the database objects before running them.
- If you use the **bpeconfig.jacl** tool to configure Business Process Choreographer, use the **-createDB** no option. This prevents the tool from running the SQL script that it generates. The generated SQL files are based on the original SQL files that are provided for your database, but with all configuration parameters that are provided to the bpeconfig.jacl tool pre-filled in the SQL file, which minimizes the customization necessary.
 - If you use the administrative console's **Business Process Choreographer configuration page** or **Deployment Environment wizard** to configure Business Process Choreographer, plan to clear the **Create tables** option, to make sure that you do not get the default schema. The generated SQL files are based on the original SQL files that are provided for your database, but all configuration parameters that you enter in the administrative console are pre-filled in the generated SQL file, which minimizes the customization necessary.

For more information about using the generated SQL scripts, refer to Using a generated script to create the database schema for Business Process Choreographer. If you want to preview the original SQL files for your database, so that you can plan what customizations you will make, locate and view the createSchema_BPC.sql script for your DB2 for z/OS database or createSchema.sql for other databases, but do not modify it. The original SQL files are located in the following directory:

- **Linux** **UNIX** `install_root/dbscripts/ProcessChoreographer/database_type`
- **Windows** `install_root\dbscripts\ProcessChoreographer\database_type`

Where `database_type` is one of the following:

- DB2
 - DB2zOS
 - Oracle
 - SQLServer
7. Decide which server will host the database. If the database server is remote, you need a suitable database client or a type-4 JDBC driver that has XA-support.
 8. If the database server is remote to the Process Server system, plan to install either a Java Database Connectivity (JDBC) driver or a database client on the Process Server system:
 - For a Type 2 JDBC driver: Decide which database client to install, and where to install it.
 - For a Type 4 JDBC driver: Locate the JAR file for the driver, which is provided as part of your product installation, and decide where to install it.
 9. If the database server is local to the Process Server system, the JDBC JAR files required to access the database are installed with the database system. Find and note the location of these JAR files.
 10. If you use DB2 for z/OS, plan the changes you will make to the script files `createTablespace_BPC.sql` and `createSchema_BPC.sql`.
 - a. Plan the values that you will substitute for storage group name, database name (not the subsystem name), and schema qualifier.
 - b. Evaluate the default assignments of tables to table spaces. Then plan any necessary changes to suit your setup.
 - c. Evaluate the default assignment of table spaces to buffer pools, and the default buffer pool sizes. Then plan any necessary changes to suit your setup.
 11. Decide the values for the following configuration parameters that you will need to specify for the database:
 - The Java Database Connectivity (JDBC) provider can be type-2 or type-4. For Oracle, only the type 4 thin driver is supported.
 - Database instance (for Oracle, the database name, for DB2 on z/OS: the subsystem name).
 - Schema qualifier. The default is to use the connection user ID as the implicit schema qualifier.
 - User name to create the schema.
 - If you are using a type-4 JDBC driver: the name or IP address of the database server.
 - The port number used by the database server. This is only required if you are using a type-4 JDBC driver.
 - The user ID and password for the authentication alias. This is the user ID that the jdbc/BPEDB data source uses to access the database at run time. These are the `-dbUser` and `-dbPwd` parameters for `bpeconfig.jacl`.
 12. Plan to support enough parallel JDBC connections:
 - a. Estimate the maximum number of parallel JDBC connection required to the Business Process Choreographer BPEDB database. This will depend on the nature of your BPEL processes and the number of users. A good estimate is the maximum number of clients that can concurrently connect through the Business Process Choreographer API plus the number of concurrent endpoints defined in the `BPEInternalActivationSpec` and `HTMInternalActivationSpec` JMS activation specifications, plus a 10% safety margin to allow for overload situations.

- b. Make sure that your database system can support the necessary number of parallel JDBC connections.
 - c. Plan suitable settings according to the best practices for your database system to support the expected number of parallel JDBC connections.
13. By default, JDBC batching for Oracle is enabled, which can improve performance. If you want to disable JDBC batching for Oracle, set the Business Flow Manager custom property **DatabaseBatchUpdate** to the value **false**. If the property is not set, the default is that JDBC batching for Oracle databases is enabled.
 14. For a production system, make plans for the following administration tasks:
 - Tune your database after it is populated with typical production data.
 - Retune your database after your application workload evolves.
 - Regularly delete completed process instances and task instances from the database. For an overview of the tools and scripts that are available, see Cleanup procedures for Business Process Choreographer.

You have planned the database for Business Process Choreographer.

Planning the messaging engine database:

For high-load setups, where database logging might become a bottleneck, you can improve performance by using a separate database for the messaging engine for the Business Process Choreographer bus. To achieve the best performance for long-running processes, tune the messaging system for maximum performance of persistent messages.

You can use the same messaging database for each messaging engine for the Service Component Architecture (SCA) system bus, each messaging engine for the SCA application bus, each messaging engine for the Common Event Infrastructure bus, and each messaging engine for the Business Process Choreographer bus. The database should be accessible to all members of the cluster that hosts the message engine to ensure failover availability of the message engine. If performance is important, plan to use a dedicated database for the Business Process Choreographer messaging engine, rather than using the default MEDB that is used for the SCA bus and applications.

1. If you use the `bpeconfig.jacl` configuration script to configure Business Process Choreographer, Business Process Choreographer will use the same database as a message store that is used by SCA. Business Process Choreographer will use its own schema in the same database.
2. If you use the administrative console's Business Process Choreographer configuration page, if you want to use the default configuration that is based on the SCA message store settings, plan to select the **Use the default configuration** check box, otherwise, plan the following configuration parameters:
 - Local or remote bus member location.
 - The name of the database. The default is `BPEME`.
 - The schema name. The default is `MEDBPM00`.
3. Plan the following configuration parameters.
 - a. Plan that the database will already exist before Business Process Choreographer is started.
 - b. The host name or IP address of the database server, and the port number that it uses.
 - c. The user name used to connect to the database and to create the schema. This is the user ID that you planned in Table 52 on page 108.

You have planned the database for the Business Process Choreographer messaging engine.

Planning the Business Process Archive database:

Each Business Process Archive Manager configuration requires its own archive database. If you have multiple Business Process Archive Manager configurations, then each of these needs its own database or database schema. The Business Process Archive database tables cannot be shared between multiple Business Process Archive Manager configurations.

1. Identify which Business Process Archive Manager this database will be used by.
2. Identify the source Business Process Choreographer configuration from which data will be archived to this Business Process Archive Manager.
3. Your archive database must be of the same type that is used by the Business Process Choreographer configuration.
 - a. Decide the name of the new database. The default name is BPARCDB.
 - b. Depending on the database type, identify the connection information such as the hostname, port number, and user ID.

Note: If you are using a DB2 for z/OS database, and you want to use the same DB2 subsystem for both the archive database and the Business Process Choreographer database, you must use a different database name and a different user ID.

You have planned the database for the Business Process Archive Manager.

Planning for the Business Flow Manager and Human Task Manager

The core of a Business Process Choreographer configuration consists of the Business Flow Manager and the Human Task Manager. You must plan their configuration parameters.

1. Make sure you know the Java Message Service (JMS) provider user ID that will be used as the run-as user ID for the Business Flow Manager message driven bean. In the administrative console, and in Table 53 on page 109, it is known as the **JMS API Authentication User**.
2. Make sure you know the Java Message Service (JMS) provider user ID that will be used as the run-as user ID for the Human Task Manager message driven bean. In the administrative console, and in Table 53 on page 109, it is known as the **Escalation User Authentication User**.
3. Make sure you know which groups or user IDs the security roles for administrator and monitor will map onto. For details, see Table 54 on page 109.
4. If you want the Human Task Manager to send email notifications of escalation events, identify the host name or IP address where the Simple Mail Transfer Protocol (SMTP) email server is located. Plan what the sender address should be for email notifications. If the email service requires authentication, make sure you know the user ID and password to use to connect to the service.
5. Decide on the context root for the web service binding of the API.
 - When configured on a server:
 - The default for the Business Flow Manager is `/BFMIF_nodeName_serverName`.
 - The default for the Human Task Manager is `/HTMIF_nodeName_serverName`
 - When configured on a cluster:
 - The default for the Business Flow Manager is `/BFMIF_clusterName`
 - The default for the Human Task Manager is `/HTMIF_clusterName`
6. If you are going to use the Business Process Choreographer Explorer, the Business Space, or a client that uses the Representational State Transfer (REST) API or the JAX web services API, decide on the context roots for the REST API and the JAX web services API.
 - The defaults for the Business Flow Manager are `/rest/bpm/bfm` and `/BFMJAXWSAPI`.
 - The defaults for the Human Task Manager are `/rest/bpm/htm` and `/HTMJAXWSAPI`.
 - When configured on a server, or on a single cluster, or on multiple clusters that are mapped to different web servers, you can use the default values.

- When configured in a network deployment environment on multiple deployment targets that are mapped to the same web server, do not use the default values. The context root for each Business Process Choreographer configuration must be unique for each combination of host name and port. You will have to set these values manually using the administrative console after configuring Business Process Choreographer.
7. Decide whether you want to initially enable audit logging for the Business Flow Manager, or Human Task Manager, or both.
 8. If you are going to use the Business Process Choreographer Explorer reporting function, decide whether you want the Business Flow Manager to be initially configured to generate Common Event Infrastructure logging events.

You have planned all the initial configuration parameters for the Business Flow Manager and Human Task Manager. You can change any of these setting anytime later using the administrative console.

Planning for the people directory provider

Plan the people directory provider, people substitution, virtual member manager, and Lightweight Directory Access Protocol (LDAP) settings for Business Process Choreographer.

1. If you are going to use human tasks, decide which people directory providers you will use:

Virtual member manager (VMM) people directory provider

The VMM people directory provider is ready to use federated repositories (also known as virtual member manager) as is preconfigured for WebSphere security - using a file repository. If you want to use a different user repository with federated repositories, you will need to reconfigure federated repositories. The VMM people directory provider supports all Business Process Choreographer people assignment features including substitution. It relies on the features provided by federated repositories, such as support for different repository types, such as LDAP, database, file based, and property extension repository.

To use the VMM people directory provider requires that you have configured federated repositories for WebSphere Application Server security. You can associate federated repositories with one or more user repositories, based on a file, LDAP, or a database. For more information about this, see *Managing the realm in a federated repository configuration*. For more information about using federated repositories, see *IBM WebSphere Developer Technical Journal*.

Lightweight Directory Access Protocol (LDAP) people directory provider

This people directory provider must be configured before you can use it. Perform the planning in step 2.

System people directory provider

This people directory provider can be used without configuring it. Do not use this provider for a production system, it is only intended for application development testing.

User registry people directory provider

This people directory provider can be used without configuring it. Depending on the WebSphere security realm definition, the user registry can use one of the following repositories:

- Federated repository - which can use the following:
 - File registry
 - One or more LDAPs
 - One or more databases
- Standalone LDAP
- Standalone custom
- Local operating system

2. If you are going to use the Lightweight Directory Access Protocol (LDAP), plan the following.

- a. You might need to customize your own version of the LDAPTransformation.xml file. For the location of that file and a list of properties that you might need to customize, see Configuring the LDAP people directory provider.
- b. Plan the following LDAP custom properties:

LDAP plug-in property	Required or optional	Description
AuthenticationAlias	Optional	The authentication alias used to connect to LDAP, for example, mycomputer/My LDAP Alias. You must define this alias in the administrative console by clicking Security > Secure administration, applications, and infrastructure > Java Authentication and Authorization Service > J2C Authentication Data . If this alias is not set or if AuthenticationType is not set to simple then an anonymous logon to the LDAP server is used.
AuthenticationType	Optional	If this property is set to simple , for simple authentication, then the AuthenticationAlias parameter is required. Otherwise, if it is not set, anonymous authentication is used.
BaseDN	Required	The base distinguished name (DN) for all LDAP search operations, for example, o=mycompany, c=us. To specify the directory root, specify an empty string using two single quotation marks, ''.
Casesentiveness ForObjectclasses	Optional	Determines whether the names of LDAP object classes are case-sensitive.
ContextFactory	Required	Sets the Java Naming and Directory Interface (JNDI) context factory, for example, com.sun.jndi.ldap.LdapCtxFactory
ProviderURL	Required	This web address must point to the LDAP JNDI directory server and port. The format must be in normal JNDI syntax, for example, ldap://localhost:389. For SSL connections, use the LDAP's URL. For a high availability configuration, where you have two or more LDAP servers that maintain mirrored data, plan to specify a URL for each LDAP server and use the space character to separate them.
SearchScope	Required	The default search scope for all search operations. Determines how deep to search beneath the baseDN property. Specify one of the following values: objectScope, onelevelScope, or subtreeScope
additionalParameter Name1-5 and additionalParameter Value1-5	Optional	Use these name-value pairs to set up to five arbitrary JNDI properties for the connection to the LDAP server.

3. If you are going to use the virtual member manager, plan the following.
 - a. You might need to customize your own version of the VMMTransformation.xml file. For the location of that file and a list of properties that you might need to customize, refer to Configuring the Virtual Member Manager people directory provider.
4. If you want to use people substitution, consider the following:
 - You must use the VMM people directory provider. The LDAP, system, and user registry people directory providers do not support people substitution.
 - If you are going to use people substitution in a production environment, plan to use the VMM Property Extension Repository to store the substitution information. The Property Extension Repository and, implicitly, the selected database must be unique and accessible from within the whole cell. As the BPEDB database is not necessarily unique within a cell, BPEDB cannot be used. You can use the common database, WPSRCDB, to host the Property Extension Repository, however, for a production environment, it is recommended to use a database that is independent of other Process Server databases.

- To use people substitution in a single-server test environment, you can store people substitution information in the internal file registry that is configured for federated repositories.

You have planned the people directory provider and people assignment options.

Planning for the Business Process Choreographer Explorer and Business Process Archive Explorer

Plan the configuration options and parameters for the Business Process Choreographer Explorer and Business Process Archive Explorer.

If you will use the Business Process Choreographer Explorer or Business Process Archive Explorer you can either configure them at the same time as you configure Business Process Choreographer or Business Process Archive Manager, or you can do it later.

The reporting function is optional, and can be used with Business Process Choreographer Explorer and Business Process Archive Explorer.

1. Decide how many Business Process Choreographer Explorer and Business Process Archive Explorer instances you want to configure. You can easily create the first instance while configuring Business Process Choreographer or Business Process Archive Manager.
 - Because each Business Process Choreographer Explorer instance can only connect to one Business Process Choreographer configuration, it makes sense to set up a Business Process Choreographer Explorer instance for each Business Process Choreographer configuration.
 - Similarly, because each Business Process Archive Explorer instance can only connect to one Business Process Archive Manager configuration, it makes sense to set up a Business Process Archive Explorer instance for each Business Process Archive Manager configuration.
 - You might want to have two or more differently customized Business Process Choreographer Explorer or Business Process Archive Explorer configurations that connect to the same Business Process Choreographer or Business Process Archive Manager. You can customize each version independently, for more information about what you can customize, see Customizing Business Process Choreographer Explorer or Business Process Archive Explorer.
2. For each Business Process Choreographer Explorer and Business Process Archive Explorer instance that you want, plan the following:
 - a. The deployment target where you will configure the instance.
 - You can configure multiple Business Process Choreographer Explorer and Business Process Archive Explorer instances on each server or cluster. Whether an instance behaves like a Business Process Choreographer Explorer or Business Process Archive Explorer actually depends on the configuration that the instance connects to.
 - You can create the instances on any deployment target.
 - b. The context root. It must be unique within the cell. The default for the Business Process Choreographer Explorer is `/bpc`. The default for the Business Process Archive Explorer is `/bpcarchive`.
 - c. The URL for the Business Process Choreographer Explorer that will, for example, be inserted in escalation e-mails.

Important: Any attempts to access the Business Process Choreographer Explorer or the Business Process Archive Explorer using the HTTP protocol are redirected to use HTTPS.
 - d. If you are planning to configure a Business Process Choreographer Explorer, plan the URL for the Business Flow Manager and Human Task Manager representational state transfer (REST) APIs endpoints. They must match the values for the context roots that you planned for the REST APIs. For example, if the context root for the Human Task Manager web service is `/rest/bpm/htm`, the endpoint URL for the Human Task Manager REST API endpoint would be `https://hostname:port/rest/bpm/htm`.
 - e. The maximum number of results to be returned for a query - the default is 10000.

- f. The deployment target (server or cluster) of the Business Process Choreographer or Business Process Archive Manager instance that this Business Process Choreographer Explorer or Business Process Archive Explorer will connect to.

You have planned the configuration options for the Business Process Choreographer Explorer and Business Process Archive Explorer.

Planning for the Business Process Archive

If you configure a Business Process Archive Manager, you can move the data associated with completed process instances and human tasks from the BPEDB database to a separate archive database.

To familiarize yourself with the Business Process Archive Manager elements, architecture, and possible configurations, read “BPEL process archiving overview” on page 90.

The Business Process Archive Manager requires its own database. To access the archived process instance and human task data, you can either use the Business Process Archive Explorer or use the Process Archive Manager EJB API to allow a custom client to access the archived data. If you do not perform any database cleanup procedures or archiving, your runtime database will steadily grow in size, which can result in a gradual deterioration in the database performance.

1. For each top-level BPEL process and top-level human task, consider the retention, archiving, and deletion requirements:
 - a. How long should they be retained in the Business Process Choreographer database before being deleted or moved to the archive?
 - b. If they will be archived, which archive will they be moved to, and who requires access to them?
 - c. Make sure that your archive policy is coordinated with your cleanup procedures to ensure that no instances that should be archived are deleted from the Business Process Choreographer database. For more information about cleanup procedures, refer to Cleanup procedures for Business Process Choreographer.
2. Decide how many Business Process Archive Manager configurations you need. Keep the following in mind:
 - Each Business Process Archive Manager configuration can only be used to archive data from one Business Process Choreographer configuration.
 - You can have multiple Business Process Archive Manager configurations that archive data from the same Business Process Choreographer configuration.
 - Federation of data between Business Process Archive databases is not a feature that is built into IBM Business Process Manager Advanced.
 - An example of when you might want multiple Business Process Archive Manager configurations is when process instances and task instances that belong to different applications are processed on the same Business Process Choreographer configuration.
3. For each Business Process Archive Manager configuration, perform “Planning for the Business Process Archive Manager” on page 123.
4. Decide how many Business Process Archive Explorer configurations you need. Unless you are going to develop your own custom client that can connect to and work with Business Process Archive Manager configurations, you probably need one Business Process Archive Explorer configuration for each Business Process Archive Manager configuration.
5. For each Business Process Archive Explorer configuration, perform “Planning for the Business Process Choreographer Explorer and Business Process Archive Explorer” on page 121.
6. If you want your own custom client to access archived process instances and human tasks, refer to Business Process Archive Manager EJB API support.

Planning for the Business Process Archive Manager:

Configuring the Business Process Archive Manager requires a database, to which completed process instances and human tasks can be moved.

1. For the Business Process Archive Manager, plan the following:
 - a. Identify the deployment target of the source Business Process Choreographer configuration that this Business Process Archive Manager will serve.
 - b. Identify the deployment target for the Business Process Archive Manager. Keep in mind that it cannot coexist on a deployment target with any Business Process Choreographer or other Business Process Archive Manager configuration. However, it must be in the same cell as the Business Process Choreographer configuration from which it will archive instances. Consider configuring it on a support cluster, if you have one.
 - c. Plan to configure the Service Component Architecture (SCA) on the deployment target of the Business Process Archive Manager before you configure run the script to configure Business Process Archive Manager. You need to know the remote bus member location of the SCA configuration that is used by the Business Process Choreographer configuration. Do not plan any other values for configuring SCA for the Business Process Archive Manager, because you must use the prefilled values for the datastore settings.
 - d. You can only configure the Business Process Archive Manager using the `bpeconfig.jacl` script. Plan all the configuration parameters. The following table contains some guidelines. For detailed information about these parameters, including default values, see `bpeconfig.jacl` script file.

Table 58. Planning `bpeconfig.jacl` parameters for a Business Process Archive Manager configuration

Condition	Planning action
Only if the deployment manager is not running:	Include the <code>wsadmin</code> option: <code>-conntype NONE</code>
If the deployment manager is running and administrative security is enabled:	Include the following <code>wsadmin</code> parameters for a user ID that is authorized to run wsadmin : <code>-user userName</code> <code>-password userPassword</code>
If you are not using the default profile:	Include the <code>wsadmin</code> parameter: <code>-profileName profileName</code>
If you want to run the script interactively and be prompted for all mandatory and optional parameters:	Invoke the script and only specify any necessary <code>wsadmin</code> parameters (listed earlier in this table), but none of the script parameters listed below in this table. By specifying no script parameters, -promptMode defaults to ASK .
If you want to run the script interactively and be prompted for all mandatory and optional parameters, but also specify some script parameters:	Specify any necessary <code>wsadmin</code> parameters and any script parameters, but also include the option: <code>-promptMode ASK</code> Any script parameters that you specify are also prompted for, and the values you provided are offered as defaults, which you can accept or change.
If you want to run the script interactively and want any missing optional script parameters to take their default values without being prompted for:	Specify any necessary <code>wsadmin</code> parameters and all mandatory script parameters, but also include the option: <code>-promptMode USE_DEFAULT</code> If a mandatory parameter is missing the script switches to -promptMode ASK mode.

Table 58. Planning `bpeconfig.jacl` parameters for a Business Process Archive Manager configuration (continued)

Condition	Planning action
If you want any missing mandatory parameters to cause the script to fail, rather than prompting for them:	<p>Include the option:</p> <pre>-promptMode FAIL_IF_MISSING</pre> <p>Use this option if the script is invoked without the possibility to get input from the console.</p>
If you want to use a response file that specifies parameter values for the script:	<p>Include the option:</p> <pre>-options responseFile</pre> <p>For more information, refer to the description of the -options parameter in <code>bpeconfig.jacl</code> script file.</p>
To create a Business Process Archive Manager configuration:	<p>Include the option:</p> <pre>-operationMode ARCHIVE</pre>
If you are not creating the configuration on the default server:	<p>Include either the parameter:</p> <pre>-cluster clusterName</pre> <p>or both parameters:</p> <pre>-node nodeName -server serverName</pre>
For every Business Process Archive Manager configuration:	<p>Include the required parameters for authorization:</p> <pre>-jmsBFMRUNAsUser userID -jmsBFMRUNAsPwd password -jmsHTMRUNAsUser userID -jmsHTMRUNAsPwd password</pre> <p>For the following administrator and monitor parameter pairs ending in <i>Users</i> and <i>Groups</i> you must specify either one or both parameters.</p> <pre>{-adminUsers userList -adminGroups groupList} {-monitorUsers userList -monitorGroups groupList}</pre> <p>The administration job user ID and password are optional.</p> <pre>-adminJobUser userID -adminJobPwd password</pre>
If you do not want to use the default context roots for the APIs:	<p>Include the optional parameters for context roots:</p> <pre>-contextRootBFMWS contextRootBFMWS -contextRootBFMREST contextRootBFMREST -contextRootHTMWS contextRootHTMWS -contextRootHTMREST contextRootHTMREST</pre>

Table 58. Planning *bpeconfig.jacl* parameters for a Business Process Archive Manager configuration (continued)

Condition	Planning action
<p>Because you can either have the script file create the Business Process Archive database, or just have it generate the SQL script without running the scripts:</p>	<p>Use the option <code>-createDB { yes no }</code></p> <p>If you select yes, the <i>bpeconfig.jacl</i> script will generate and run an SQL file to create the database objects in the default table space, which is not suitable for a high-performance system. In this case also plan to stop the deployment manager and use the <code>-conntype NONE</code> option.</p> <p>If you select no, and the database does not already exist, then you or your database administrator must run the generated SQL script. For a high-performance system, specify no, because you will need to customize the SQL script before running it. Also specify no if you do not have the authority to create the database yourself, so that you can provide the SQL script to your database administrator to customize and run.</p> <p>You must also specify no if you are using a database which has restricted support. Restriction: The script cannot create the following types of databases:</p> <ul style="list-style-type: none"> • DB2 for z/OS • Oracle • A remote Microsoft SQL Server <p>If you select yes and you are running the script in connected mode, creating the database or schema can fail if it takes longer than the a default timeout of 3 minutes. How to change the timeout is described in Connection timeout when running a wsadmin script .</p>
<p>Because every Business Process Archive Manager configuration requires a database:</p>	<p>Include the parameter: <code>-dbType <i>databaseType</i></code></p> <p>Also provide the parameters required for your database type (see <i>bpeconfig.jacl</i> script file for details):</p> <pre>-dbVersion <i>version</i> -dbHome <i>databaseInstallPath</i> -dbJava <i>JDBCdriverPath</i> -dbName <i>databaseName</i> -dbUser <i>databaseUser</i> -dbPwd <i>databasePassword</i> -dbTablespaceDir <i>databaseTablespacePath</i> -dbServerName <i>databaseServerName</i> -dbServerPort <i>databaseServerPort</i> -dbStorageGroup <i>DB2zOSSStorageGroup</i> -dbConnectionTarget <i>DB2zOSSubSystem</i> -dbSchema <i>schemaQualifier</i></pre> <p>When running the script in batch mode on a cluster, if your database requires the <code>-dbJava</code> parameter, specify the parameter for each node that hosts a cluster member in the following way:</p> <pre>-dbJava.<i>nodeName</i> <i>JDBCdriverPath</i> _on_<i>nodeName</i></pre> <p>Note: If you are using one of the following databases, <i>bpeconfig.jacl</i> can also create the database instance:</p> <ul style="list-style-type: none"> • A local DB2 for Linux, UNIX, or Windows

Table 58. Planning *bpeconfig.jacl* parameters for a Business Process Archive Manager configuration (continued)

Condition	Planning action
If you do not want a Business Process Archive Explorer on the same deployment target as the Business Process Archive Manager:	<p>Include the parameter:</p> <pre>-createExplorer no</pre> <p>Note: You can run the <i>clientconfig.jacl</i> script later to create Business Process Archive Explorer configurations.</p>
If you want a Business Process Archive Explorer on the same deployment target as the Business Process Archive Manager:	<p>Include the following parameters.</p> <pre>-createExplorer yes -restAPIBFM restAPIURL -restAPIHTM restAPIURL</pre> <p>Also include any of the following optional parameters:</p> <pre>-contextRootExplorer explorerContextRoot -hostName explorerVirtualHostname -maxListEntries maximum -remoteCluster clusterName -remoteNode nodeName -remoteServer serverName</pre> <p>If you do not specify these parameters, the default values are used.</p>
Because not all <i>bpeconfig.jacl</i> parameters apply to the Business Archive Manager configuration	<p>Do not plan values for the following parameters:</p> <pre>-explorerHost -mailPwd -mailServerName -mailUser -medbPwd -medbUser -mqCreateTables -mqPwd -mqSchemaName -mqUser</pre> <p>If you provide any of these parameters, they are ignored.</p>

2. If you have not already planned the process archive database for this Business Process Archive Manager, perform “Planning the Business Process Archive database” on page 118.

You have planned a Business Process Archive Manager configuration and its database.

Planning for a remote client application

Planning for a remote Business Process Choreographer client application that uses the Business Process Choreographer APIs and runs on an IBM BPM Advanced client installation.

If you want an application to use the Business Process Choreographer APIs, you can use an IBM BPM Advanced client installation to run the applications remotely against a full IBM BPM Advanced installation. The client is easier to configure and administer than a full IBM BPM Advanced installation.

You can even install the IBM BPM Advanced client on top of an existing WebSphere Application Server installation that has federated profiles and those federated WebSphere Application Server profiles can take advantage of the IBM BPM Advanced client functionality immediately. This scenario is not possible with the full IBM BPM Advanced because IBM Business Process Manager Advanced does not support augmentation of already federated profiles.

1. Plan to install an IBM BPM Advanced client.

- You can you can install the client on servers and clusters in standalone and managed profiles. If you want to use the IBM BPM Advanced client in a cluster, then you must install the IBM BPM Advanced client on all WebSphere Application Server installations that host cluster members.
 - If you want WebSphere Portal Server to access Business Process Choreographer, you must have a compatible IBM BPM Advanced client installed. For more information about compatibility between IBM BPM Advanced and WebSphere Portal, refer to the Technote <http://www.ibm.com/support/docview.wss?uid=swg21580260>.
 - Any existing profiles, including already federated profiles, can use the IBM BPM Advanced client immediately, because the client installation does not augment the base profile.
 - If there is no existing WebSphere Application Server installation, a WebSphere Application Server network deployment installation will be created.
2. Decide which type of Business Process Choreographer client application you will use:
 - Custom client application
 - Business Process Choreographer Explorer

Note: If you use customized JavaServer Pages (JSP), as described in Developing JSP pages for task and process messages, make sure that you know where they are located.
 3. If you are going to develop a custom client application that will use Business Process Choreographer, plan which interfaces the application will use. You can handle processes and tasks using one of the following:
 - Web services API, Java Messaging Service (JMS) API or representational state transfer (REST) API – remote client applications that are based on these APIs do not need any WebSphere Process Server installation.
 - JavaServer Faces (JSF) components
 - Enterprise JavaBeans (EJB) API

Note: If you develop a client application, which uses the Business Process Choreographer EJB APIs, it must be packaged in the way that is described in Accessing the remote interface of the session bean.
 4. Decide or identify the type of cell where the IBM BPM Advanced client will be installed:
 - a. In a cell where a managed server or cluster is located, on which Business Process Choreographer is configured, the default configuration of the Remote Artifact Loader (RAL) allows the unsecured transmission of artifacts between the client and the server. This is known as the “single-cell” scenario.
 - b. In a cell that does not have a managed server or cluster with Business Process Choreographer configured on it, there are different deployment managers. This is known as the “cross-cell” scenario. If your client application uses the EJB API, you must define a namespace binding so that the client application can locate the server or cluster where Business Process Choreographer is configured.

You have planned for a remote Business Process Choreographer client application.

Chapter 5. Installing and configuring IBM Business Process Manager Advanced

IBM Business Process Manager Advanced can be installed and configured on a single server (known as a stand-alone configuration), or you can distribute the components across multiple systems (known as a network deployment configuration).

Roadmap: Installing and configuring IBM Business Process Manager Advanced

The IBM Business Process Manager Advanced installation and configuration programs accommodate various scenarios. This roadmap guides you through the most common paths for installing and configuring IBM Business Process Manager Advanced in a stand-alone or network deployment environment.

Alternatively, you can use the Interactive Installation and Configuration Guide to answer a series of questions about your target environment. The guide generates a set of installation and configuration topics that are customized to your installation needs.

The roadmap provides the most relevant links for each common installation and configuration scenario, grouped by operating system.

View all | View with tabs

- “Linux” on page 1
- “AIX” on page 2
- “Solaris” on page 2
- “Windows” on page 3

Linux

To install IBM Business Process Manager on Linux, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing Linux systems for installation” on page 135

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 142
- “Custom installation and configuration” on page 166

- “Installing IBM Business Process Manager Advanced interactively” on page 166
- “Installing IBM Business Process Manager silently” on page 175

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 182

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 605
- Creating a network deployment environment

AIX

To install IBM Business Process Manager on AIX, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing AIX systems for installation” on page 134

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 1002
- “Custom installation and configuration” on page 1022
 - “Installing IBM Business Process Manager Advanced interactively” on page 1022
 - “Installing IBM Business Process Manager silently” on page 1026

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 1032

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 1453
- Creating a network deployment environment

Solaris

To install IBM Business Process Manager on Solaris, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing Solaris systems for installation” on page 137

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 1846
- “Custom installation and configuration” on page 1866
 - “Installing IBM Business Process Manager Advanced interactively” on page 1866
 - “Installing IBM Business Process Manager silently” on page 1870

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 1876

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 2300
- Creating a network deployment environment

Windows

To install IBM Business Process Manager on Microsoft Windows, follow this roadmap.

Preparing

Complete all the following tasks: review the hardware and software requirements, prepare the operating system for installation, and ensure that you have installed the database management system.

- “System requirements” on page 133
- “Preparing Windows systems for installation” on page 137

Deciding

Before proceeding with the installation, determine which type of installation you want to perform, typical or custom, and choose the appropriate deployment environment for your installation, stand-alone or network deployment.

- “Choosing the type of installation” on page 24
- Choosing a stand-alone or network deployment environment

Installing

Based on the decisions made earlier, select the installation option that is most appropriate for your deployment environment, and then complete the steps to install the product.

- “Typical installation and configuration” on page 2697
- “Custom installation and configuration” on page 2722
 - “Installing IBM Business Process Manager Advanced interactively” on page 2723
 - “Installing IBM Business Process Manager silently” on page 2732

Configuring

No configuration is required after the completion of a typical installation. Perform the following configuration after a non-typical installation.

Stand-alone deployment

“Configuring profiles and databases” on page 2739

Network deployment

Based on your configuration, choose the database that you want to work with, and follow the steps to configure the profiles and build the network deployment configuration.

- “Configuring profiles and creating a network deployment environment” on page 3169
- Creating a network deployment environment

Preparing to install and configure the software

Before preparing to install and configure the software, create a plan for the deployment environment that you want to create.

Use the information listed in the following table to prepare for installing and configuring IBM Business Process Manager.

Table 59. Preparing for installation and configuration

Task	Where to find information	Result after completing the task
Review hardware and software requirements	Depending on your IBM BPM configuration, visit: <ul style="list-style-type: none">• IBM Business Process Manager Advanced system requirements• IBM Business Process Manager Standard system requirements• IBM Business Process Manager Express system requirements	You understand the system requirements necessary to support your IBM Business Process Manager installation.
Prepare your operating system	<ul style="list-style-type: none">AIX Preparing AIX systems for installationLinux Preparing Linux systems for installationSolaris Preparing Solaris systems for installationWindows Preparing Windows systems for installation	You have prepared the operating system of each workstation to be used.

Table 59. Preparing for installation and configuration (continued)

Task	Where to find information	Result after completing the task
<p>Make sure you have installed your database management system.</p>	<p>Consult your database documentation for information about installing and administering your database management system.</p>	<p>Your database management system is installed.</p> <p>IBM Business Process Manager embeds the DB2 Express database. If you want to use DB2 Express as your database, you can select it as a feature from the installer and it is installed and configured automatically. The user must have administrative privileges (root or Administrator) to install DB2 Express.</p> <p>Note: If you already have a version of DB2 installed and you want to install DB2 Express, you must uninstall DB2 before running the IBM Business Process Manager installer. If the installer detects a version of DB2 installed and you have selected to install DB2 Express from the installer, you will receive a warning message and will not be able to install DB2 Express.</p> <p>Important: Linux If you are installing DB2 Express as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. See Kernel parameter requirements (Linux) for a list of the kernel requirements. You can locate the current values by parsing the output of the <code>ipcs -l</code> command.</p> <p>Important: You will not be able to properly install and use DB2 Express if the password that is specified does not meet operating system or company restrictions, because an operating system user is created on install.</p>

System requirements

Before you install, ensure that your system meets all system requirements.

For the latest information about platform-specific disk space requirements, supported operating systems, and supported database versions, click one of the following links. You can also find operating system fixes and patches that you must install to have a compliant operating system.

- IBM Business Process Manager Advanced system requirements
- IBM Business Process Manager Tools and Add-Ons Requirements

Preparing operating systems for product installation

Before you can install IBM Business Process Manager, you must prepare your operating system. The configuration depends on the type of operating system you are using.

Before preparing the installation environment, complete the following tasks:

- Disable the firewall if you have a firewall running on the system where you plan to install IBM Business Process Manager.
- Ensure that your user login provides access to your DB2 or Oracle database commands.

- Complete additional tasks specific to your operating system.

Attention: While installing IBM Business Process Manager on **Windows Server 2003** or **Windows 7 (English)** and switching the locale to Czech in the IBM Process Portal **Preference -> Interface language** the text displays with corrupted characters on several panels of the IBM Process Designer. The text displays with corrupted characters even if IBM Process Designer is started with the Czech locale.

Tip: To resolve this issue change system settings as follows.

- In **Windows Server 2003: Regional and Language Options -> Advanced -> Language for non-Unicode programs -> set to 'Czech'**
- In **Windows 7: Regional and Language Options -> Administrative Tab -> Change system locale... (under 'Language for non-Unicode programs') -> Select 'Czech'**

Preparing AIX systems for installation

Before you can install IBM Business Process Manager, you must prepare your AIX operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete the required preparation steps in the Preparing AIX systems for installation topic in the WebSphere Application Server information center.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Refer to the following technote for additional preparation information for configuring Installation manager to run on 64-bit AIX systems: <https://www-304.ibm.com/support/docview.wss?uid=swg21330190&wv=1> .

Complete the following steps on your AIX system before installing IBM Business Process Manager:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using **ulimit -n**. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The **ulimit** requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

Before installing, run the following command:

```
ulimit -n 8800
```

Alternatively, you can use the following steps to edit the resource limits file:

- a. Open `/etc/security/limits`.
- b. Edit or add the **default** section and include this line:
nfiles = 8800
- c. Save and close the file.
- d. Log off from the operating system and log in again.

2. Set the **umask** value to 077 using the following command:

```
umask 077
```

The value 077 is the most restrictive value that IBM Business Process Manager will tolerate. You can optionally choose to set a less restrictive **umask** value for the following access levels:

- 037 for read-only access for a group of human administrators and tools
- 027 for read and write access for a group of human administrators and tools
- 007 for read, write, and execute access for a group of human administrators and tools

3. Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.

4. Before starting the data movement service, increase the number of processes configured in the AIX operating system to avoid a connection reset error. You can increase the number of processing using a command, or using the AIX interface.
 - Run the command:


```
chdev -l sys0 -a maxuproc='256'
```
 - In the AIX interface, enter **smitty**, then select **System Environments > Change / Show Characteristics of Operating System > Number of processes allowed per user(Num.)**.
5. Complete the steps in Tuning AIX systems.
6. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Related tasks:

 [Preparing AIX systems for installation](#)

 [Tuning AIX systems](#)

Preparing Linux systems for installation

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete all the required preparation steps in the Preparing Linux systems for installation topic in the WebSphere Application Server information center.

Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system. To install Installation Manager on Red Hat Enterprise Linux 6.0 (64-bit), see [Unable to install Installation Manager on RHEL 6.0 \(64-bit\)](#).

Note: Cognos bundles an alternate version of the `libfreebl3.so` library for Red Hat Enterprise Linux 6.0 (64-bit). Ensure that you preinstall this system library that is required for other utilities.

Complete the following steps on your Linux system before installing IBM Business Process Manager:

1. Increase the allowable stack size, number of open files, and number of processes by adding the following lines to the end of the `/etc/security/limits.conf` file:

```
# - stack - max stack size (KB)
* soft stack 32768
* hard stack 32768
# - nofile - max number of open files
* soft nofile 65536
* hard nofile 65536
# - nproc - max number of processes
* soft nproc 16384
* hard nproc 16384
```

Save and close the file, and log off and log in again. You can check the current maximum number of open files by using `ulimit -n`. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select. For more information about this setting, run `man limits.conf` or see the topic [Preparing the operating system for product installation](#) in the WebSphere Application Server information center.

2. Check for the existence of a file named `/etc/security/limits.d/90-nproc.conf`, which overrides the `nproc` value set in the `limits.conf` file. If the `90-nproc.conf` file exists, edit it and set the `nproc` values that are specified in the previous step.

- If you are planning to install DB2 Express as part of the installation process, ensure that you have administrative (root) privileges. As a root user, you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the `ipcs -l` command. To change the values, add the following lines, in the following order, to the `/etc/sysctl.conf` file:

```
kernel.shmmni=4096
kernel.shmmax=4294967296
kernel.shmall=8388608
#kernel.sem=<SEMMS><SEMMNS><SEMOPM><SEMMNI>
kernel.sem=250 256000 32 4096
kernel.msgmni=16384
kernel.msgmax=65536
kernel.msgmnb=65536
```

Reboot the system

- Install the following packages for your operating system:

Option	Description
Red Hat Enterprise Linux 5	libXp-1.0.0-8 rpm-build-4.4.2-37.el5
Red Hat Enterprise Linux 6	ksh-version.rpm Korn shell

You can also install a later release of any of these packages if there are new packages as errata. If you have additional packages that are specific to your hardware, install them.

The following command example shows how to use the default package manager on supported Linux distributions.

- Red Hat Enterprise Linux 5:**

```
yum install libXp rpm-build
```

- Set the `umask` value to 077 using the following command:

```
umask 077
```

The value 077 is the most restrictive value that IBM Business Process Manager will tolerate. You can optionally choose to set a less restrictive `umask` value for the following access levels:

- 037 for read-only access for a group of human administrators and tools
- 027 for read and write access for a group of human administrators and tools
- 007 for read, write, and execute access for a group of human administrators and tools

- Confirm that the required ports are available. If you install Lotus Forms Turbo on a Linux system as part of an IBM Business Process Manager installation, you need to leave the following ports free: 2809, 2810, 8085, 8880, 8881, 9043, 9044, 9402, 9403, 9405, 9406, 9060, 9061, 9443, 9444, 9080, 9100, 50000, and 55000.

Turbo installs a DB2 database unless you configure it to use an existing database. That database must use port 50000. In an advanced installation on Linux SUSE, the installation tries to assign a port other than 50000. You need to override that generated port number and manually assign port 50000. If you just accept the incorrect DB2 port and click Next, the installation will fail.

- On Red Hat Enterprise Linux 5 systems, disable SELinux, or set it to a permissive mode.
- Restart the computer.
- Complete the steps in Tuning Linux systems.
- Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Related tasks:

-  [Preparing Linux systems for installation](#)

 [Tuning Linux systems](#)

Related reference:

 [Unable to install Installation Manager on RHEL 6.0 \(64-bit\)](#)

Preparing Solaris systems for installation

Before you can install IBM Business Process Manager, you must prepare your Solaris operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete the required preparation steps in the Preparing Solaris systems for installation topic in the WebSphere Application Server information center.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Complete the following steps on your Solaris systems before installing IBM Business Process Manager:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using `ulimit -n`. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

Before installing, run the following command:

```
ulimit -Hn 8800
```

Alternatively, you can use the following steps to edit the resource limits file:

- a. Open `/etc/system`
- b. Add the following line to the end of the file:

```
set rlim_fd_max=8800
```
- c. Save and close the file.
- d. Log off from the operating system and log in again.

2. Set the `umask` value to 077 using the following command:


```
umask 077
```

The value 077 is the most restrictive value that IBM Business Process Manager will tolerate. You can optionally choose to set a less restrictive `umask` value for the following access levels:

- 037 for read-only access for a group of human administrators and tools
- 027 for read and write access for a group of human administrators and tools
- 007 for read, write, and execute access for a group of human administrators and tools

3. Complete the steps in [Tuning Solaris systems](#).
4. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Related tasks:

 [Preparing Solaris systems for installation](#)

 [Tuning Solaris systems](#)

Preparing Windows systems for installation

Before you can install IBM Business Process Manager, you must prepare your Windows operating system.

If you are planning to use DB2 Express with your IBM Business Process Manager installation, the user account must have administrative privileges (Administrator) on the machine where you will perform the installation.

Because WebSphere Application Server is a prerequisite product for IBM Business Process Manager, you must complete all of the preparation tasks for WebSphere Application Server before installing IBM Business Process Manager.

Complete the following steps on your Windows system before installing IBM Business Process Manager:

1. Confirm that the required ports are available. If you install Lotus Forms Turbo on a Windows system, you need to leave the following ports free: 2809, 2810, 8085, 8880, 8881, 9043, 9402, 9403, 9405, 9406, 9060, 9080, 9100, 50000, and 55000. To check which ports are in use, open a command window and enter the command **netstat -b** to see which executable file is using a port.
2. Complete the steps in the Preparing Windows systems for installation topic in the WebSphere Application Server information center.
3. Complete the steps in Tuning Windows systems.
4. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Related tasks:

 [Preparing Windows systems for installation](#)

 [Tuning Windows systems](#)

Considerations for HADR setup and configuration

Review the following considerations while you plan to set up and configure DB2 for high availability disaster recovery (HADR).

- “Verify status of databases for takeover”
- “Configure data sources for HADR” on page 139
- “Set up programmatic transaction retry properties” on page 139
- “Specify user roles for HADR” on page 139
- “Start the messaging engine after the takeover” on page 140
- “Access additional information” on page 140

Verify status of databases for takeover

When a site failure occurs, HADR enables the standby database to take over as the primary database, with full DB2 functionality. Before the takeover occurs, verify the DB2 HADR status of both the primary database and the standby database to see if the takeover is allowed or if an error is returned. Sometimes, the takeover might cause an error or might not be allowed.

To check the DB2 HADR status, use the **db2pd** command. For example, if you configured a database named MEDB9 with HADR, use the following command to check the status:

```
db2pd -hadr -db MEDB9
```

To check the status of all databases, use the following command:

```
db2pd -hadr -all dbs
```

For more information, see the TAKEOVER HADR command topic. Review the command parameters and the usage notes to determine whether to perform the takeover.

Configure data sources for HADR

Configure the DB2 data sources for HADR in the IBM Business Manager admin console, as described in the Configuring client reroute for applications that use DB2 databases topic.

Keep in mind the following considerations when you configure the data sources for IBM Business Process Manager products:

- It is recommended to enable the automatic client reroute when setting up DB2 HADR.
- For messaging engine and cell level data sources, the *Alternate server names* and *Alternate port numbers* properties must be configured.
- The alternate port must be the external port for the DB2 server, not the HADR service port that is specified in `/etc/services`. For example, if the HADR service port `DB2_HADR_1`, specified in `/etc/services`, is 55001, and the DB2 server uses the external port 50000, you must specify port 50000 for the data source alternate port in the admin console.

Set up programmatic transaction retry properties

To support programmatic transaction retries, two new properties have been added to the `00static.xml` file, with the following values:

```
<transaction-reroute-retries>3</transaction-reroute-retries>
<transaction-reroute-retry-delay-in-millis>10000</transaction-reroute-retry-delay-in-millis>
```

Use the `100Custom.xml` configuration file to customize the changes in `00Static.xml` if values other than default are required. Complete the following steps:

1. Locate `100Custom.xml` configuration file:
 - For a network deployment environment, the file path is `DMGR_profile_root\config\cells\cell_name\nodes\custom_node_name\servers\server_name\server_type\config\100Custom.xml`.
 - For a stand-alone server environment, the file path is `standalone_profile_root\config\cells\cell_name\nodes\standalone_node_name\servers\server_name\server_type\config\100Custom.xml`.
2. Edit the `100Custom.xml` file. Edit the following section to look like this example:

```
<server merge="mergeChildren">
  <transaction-reroute-retries merge="replace">10</transaction-reroute-retries>
  <transaction-reroute-retry-delay-in-millis merge="replace">3000
</transaction-reroute-retry-delay-in-millis>
</server>
```
3. Save your changes.

Provide all the client reroute properties for all your data sources.

Specify user roles for HADR

When setting up HADR, specify appropriate user roles per platform to determine who can perform different HADR tasks.

- For Windows, use the DB2 administrative user: `db2admin`
- For Linux, use the DB2 instance user, `db2inst1`, rather than the administrative user, `dasusr1`.

For example, if the DB2 instance user is specified, the `db2inst1` user backs up the primary database, copies the backup image to the standby server, and then restores or starts the standby server using this image. If the administrative user is specified, the `dasusr1` user is the owner of the copied image, which the `db2inst1` user who performs the backup and restore actions cannot access. Because the two users belong to different groups and have different access rights to files, the HADR setup might fail.

Start the messaging engine after the takeover

If the auto restart for the messaging engine is disabled, you must manually start the messaging engine after the database takeover occurs.

When two messaging engine servers exist in the same cluster, with messaging server 1 in an active state and messaging engine server 2 in a joined state, you might not have a running messaging engine server in your environment because of the following sequence of events:

1. The database that the messaging engines use has been taken over.
2. Messaging engine server 1 shut down to prevent data loss.
3. Messaging engine server 2 started all the messaging engines and is working as the active server while the messaging engine server 1 is still down.
4. Another takeover of the database occurred.
5. Messaging engine server 2 shut down to prevent data loss.
6. Both messaging engine servers 1 and 2 are down.

Access additional information

If more information is needed, use the following links.

- If an outage occurs in your DB2 HADR environment, see [Detecting and responding to system outages in a high availability solution](#).
- After the failover occurs, if the DB2 HADR status is not peer and if some data loss occurs, configure the DB2 HADR status to peer as described in the following topics:
 - [Performing an HADR failover operation](#)
 - [Reintegrating a database after a takeover operation](#)

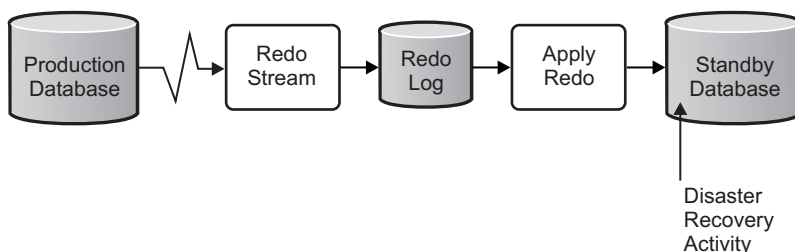
Configuring Oracle Data Guard for IBM Business Process Manager

You can configure Oracle Data Guard to be used with IBM Business Process Manager. Oracle Data Guard provides high availability, disaster recovery, and data protection and is used to create, manage, and monitor one or more standby databases so that production Oracle databases can survive disasters and data corruptions.

When IBM Business Process Manager is configured to use Oracle Data Guard, you typically have one production database, which is your primary database, and several standby databases. Then Oracle Data Guard automatically maintains each standby database by transmitting redo data from the primary database and applying the redo data to the standby database. If your production database becomes unavailable because of a planned or an unplanned outage, Oracle Data Guard enables you to switch any standby database to the production role, minimizing the downtime that is associated with the outage.

Oracle Data Guard automatically maintains each standby database by transmitting redo data from the primary database and then applying the redo to the standby database.

A Typical Data Guard Configuration



1. Set up the Oracle Data Guard environment.

2. Create a database service from the primary database:

```
Exec DBMS_SERVICE.CREATE_SERVICE('BPM','BPM');
```

3. Create a trigger from the primary database by running the following command:

```
CREATE OR REPLACE TRIGGER START_SERVICES AFTER STARTUP ON DATABASE
DECLARE
  ROLE VARCHAR(30);
BEGIN
  SELECT DATABASE_ROLE INTO ROLE FROM V$DATABASE;
  IF ROLE = 'PRIMARY' THEN
    DBMS_SERVICE.START_SERVICE('ORCL'); END IF;
END;
/
```

Note: When you start a database, your primary database always starts a BPM service. Therefore, your client always connects to the primary database.

4. Restart the primary database or start the following service by running the following command:

```
EXEC DBMS_SERVICE.START_SERVICE('BPM');
```

5. Install the IBM Business Process Manager.

6. Create profiles or a deployment environment that is similar to that of an Oracle instance database. While creating the profile, select **Oracle database** and connect to the database service that you created in step 2.

If all your primary and the standby databases are installed on the same server, the databases share the same IP address and Oracle listening port. No further configuration is required.

However, if your primary and standby databases are installed on different servers, they will have different IP addresses and the same Oracle listening port. If your primary and standby databases are on separate servers, you must modify all data sources, after which you will have a JDBC URL that resembles the following URL:

To modify the JDBC URL:

1. Log in to the IBM Business Process Manager administrative console.
2. Go to **Resources > JDBC > Data sources**.
3. Modify all data sources that connect to the Oracle database with the URLs that resemble the following URL:

```
jdbc:oracle:thin:@(DESCRIPTION=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=TCP)(HOST=<host A>)(PORT=1521))
    (ADDRESS=(PROTOCOL=TCP)(HOST=<host B>)(PORT=1521))
    (ADDRESS=(PROTOCOL=TCP)(HOST=<host C>)(PORT=1521))
    (LOAD_BALANCE=off)
    (FAILOVER=on)
  )
  (CONNECT_DATA=(SERVICE_NAME=BPM))
)
```

Installing and configuring IBM Business Process Manager Advanced

You can install IBM Business Process Manager using typical, custom, or silent installation. You can configure IBM Business Process Manager in a stand-alone or network deployment environment.

Installing and configuring IBM Business Process Manager on Linux

Install IBM Business Process Manager on Linux, and configure a stand-alone or network deployment environment.

Configuring a stand-alone environment on Linux

You can use a typical installation to install the software and configure a stand-alone Process Center or Process Server profile. Or you can choose custom installation and then create profiles, giving you more control over how to configure your software.

Typical installation and configuration:

The Typical installation option is the simplest and quickest method for installing and configuring IBM Business Process Manager.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Related tasks:

“Preparing Linux systems for installation” on page 135

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing and configuring IBM Business Process Manager with a new DB2 Express database server:

The typical installation can install DB2 Express on Linux and configure the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (root user) and do not have an existing DB2 database server on the system.

Installing and configuring Process Center with a new DB2 Express database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

The typical installation installs DB2 Express and configures the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (root user) and do not have an existing DB2 database server on the system.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

extract_directory/launchpad.sh

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **No** to install DB2 Express.
10. Specify the following DB2 administrative user names and passwords:
 - DB2 instance user
 - DB2 fenced user
 - DB2 administration server (DAS) user

If the fenced user or administration (DAS) server user name exists, clear the **New User** check box.

Restriction: User names must not contain National Language Strings (NLS).

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional

required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with a new DB2 Express database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

The typical installation installs DB2 Express and configures the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (root user) and do not have an existing DB2 database server on the system.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:

- **Hostname:** This field shows the name of your machine.
- **Location:** Click **Browse** to change the installation location.

Restriction: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

- Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
- Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Click **Next**.
8. Select **No** to install DB2 Express.
9. Specify the following DB2 administrative user names and passwords:
 - DB2 instance user
 - DB2 fenced user
 - DB2 administration server (DAS) user

If the fenced user or administration (DAS) server user name exists, clear the **New User** check box.

Restriction: User names must not contain National Language Strings (NLS).

10. Specify the following DB2 administrative user names and passwords:
 - DB2 instance user
 - DB2 fenced user
 - DB2 administration server (DAS) user

If the fenced user or administration (DAS) server user name exists, clear the **New User** check box.

Restriction: User names must not contain National Language Strings (NLS).

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with a DB2 database server:

You can install IBM Business Process Manager using a DB2 database server. You must make sure the required databases are available before you install.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
```



```
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECON 64 DEFERRED;
connect reset;
```

Installing and configuring Process Center with a DB2 database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for `ifixes`, `fix packs`, and `launchpad upgrades`. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.

4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
 8. Click **Next**.
 9. Select **Yes** to use an existing database.
 10. Specify the required database information.

Table 60. Required database configuration fields for DB2

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 50000 or enter the correct server port number.
Username	Enter the user name to authenticate with the database. Restriction: User names must not contain National Language Strings (NLS).
Password	Enter a password to authenticate with the database.
Common database name	Enter the name for the Common database.
Process Center database name	Enter the name for the Process Center database.
Performance Data Warehouse database name	Enter the name for the Performance Data Warehouse database.

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:



Installation Manager updates

Installing and configuring Process Server with a DB2 database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do

not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.

4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 61. Required database configuration fields for DB2

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 50000 or enter the correct server port number.

Table 61. Required database configuration fields for DB2 (continued)

Field	Action needed
Username	Enter the user name to authenticate with the database. Restriction: User names must not contain National Language Strings (NLS).
Password	Enter a password to authenticate with the database.
Common database name	Enter the name for the Common database.
Process Server database name	Enter the name for the Process Server database.
Performance Data Warehouse database name	Enter the name for the Performance Data Warehouse database.

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 [Installation Manager updates](#)

Installing and configuring IBM Business Process Manager with an Oracle database server:

You can install IBM Business Process Manager using an Oracle database server. You must make sure the required databases are available before you install.

Creating Oracle databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

Installing and configuring Process Center with an Oracle database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.

5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
 8. Click **Next**.
 9. Select **Yes** to use an existing database.
 10. Specify the required database information.

Table 62. Required database configuration fields for Oracle

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1521 or enter the correct server port number.
Database name	Enter the correct Process Center database System Identifier.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Center database	For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.

Table 62. Required database configuration fields for Oracle (continued)

Field	Action needed
Service Component Architecture application bus messaging engine database	<p>For the SCA application bus messaging engine, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 [Installation Manager updates](#)

Installing and configuring Process Server with an Oracle database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.

- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.


```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 63. Required database configuration fields for Oracle

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1521 or enter the correct server port number.
Database name	Enter the correct Process Center database System Identifier.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.

Table 63. Required database configuration fields for Oracle (continued)

Field	Action needed
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with an SQL Server database server:

You can install IBM Business Process Manager using a Microsoft SQL Server database server. You must make sure the required databases are available before you install.

Creating SQL Server databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
```

```
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
    at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
 - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
 - c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the `Binn` directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the `x86` folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the `x64` folder.
 - d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for `lombardi` user add master database in User mappings and check `SqlJDBCXAUser` role).

Installing and configuring Process Center with an SQL Server database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.


```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of `localhost` or `127.0.0.1` is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center.
6. Specify the **Username** and **Password** for this server's administrative account.
7. Click **Next**.
8. Select **Yes** to use an existing database.
9. Specify the required database information.

Table 64. Required database configuration fields for SQL Server

Field	Action needed
Hostname	Accept the default value of <code>localhost</code> or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1433 or enter the correct server port number.
Common database name	Enter the name for the Common database.
Process Center database name	Enter the Process Center database name.
Performance Data Warehouse database name	Enter the Performance Data Warehouse database name.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.

Table 64. Required database configuration fields for SQL Server (continued)

Field	Action needed
Process Center database	For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

10. Click **Next**.

11. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional

required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

12. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with an SQL Server database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 65. Required database configuration fields for SQL Server

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1433 or enter the correct server port number.

Table 65. Required database configuration fields for SQL Server (continued)

Field	Action needed
Common database name	Enter the name for the Common database.
Process Server database name	Enter the Process Server database name.
Performance Data Warehouse database name	Enter the Performance Data Warehouse database name.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 [Installation Manager updates](#)

Custom installation and configuration:

Use the Custom installation option to install IBM Business Process Manager if you need any installation or configuration options that are not provided by the Typical installation option, if you want to install silently, or if you want to install on an existing installation of WebSphere Application Server.

Related tasks:

“Preparing Linux systems for installation” on page 135

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced:

With custom installation, you can select interactive or silent installation. You must select interactive installation if you want to install on an existing WebSphere Application Server.

Related tasks:

“Preparing Linux systems for installation” on page 135

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced interactively:

You can install the IBM Business Process Manager product package interactively. You must choose interactive installation if you want to install on an existing WebSphere Application Server.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related tasks:

“Preparing Linux systems for installation” on page 135

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing interactively with a new installation of WebSphere Application Server:

Using custom installation, you can install IBM Business Process Manager with a new installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage®. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```

- If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
5. Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
6. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package.

7. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

8. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.
If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.
10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
12. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:



Installation Manager updates

Installing interactively with a new DB2 Express database and a new installation of WebSphere Application Server:

The custom installation can install DB2 Express on Linux and configure the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (root user) and do not have an existing DB2 database server on the system.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```

- If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
5. Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
6. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Select IBM DB2 Express to install and use an embedded DB2 Express database.
7. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

8. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Attention: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. Specify the DB2 administrative user names and passwords on the Common Configurations page. The default user names are:
 - Instance user name: bpminst.
 - Fenced user name: bpmfenc
 - Administration server (DAS) user name: bpmadmin

If the fenced user or administration (DAS) server user name exists, clear the **New User** check box. Click **Re-Validate** to confirm that your choices are correct.

Restriction: User names must not contain National Language Strings (NLS).

12. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back**

and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.

13. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 [Installation Manager updates](#)

Installing interactively on an existing installation of WebSphere Application Server:

Using custom installation, you can install IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Installation on an existing WebSphere Application Server**.

5. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
6. Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
7. Click **Install**.
8. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server.
9. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

10. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
11. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.

14. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 Installation Manager updates

Installing interactively with a new DB2 Express database on an existing installation of WebSphere Application Server:

Using custom installation, you can install DB2 Express and IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:


```
mount_point/launchpad.sh
```
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Installation on an existing WebSphere Application Server**.

5. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
6. Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
7. Click **Install**.
8. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server. Select IBM DB2 Express to install and use an embedded DB2 Express database. The user must have administrative privileges to install DB2 Express.
9. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

10. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
11. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Attention: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. Specify the DB2 administrative user names and passwords on the Common Configurations page. The default user names are:
 - Instance user name: bpminst.
 - Fenced user name: bpmfenc
 - Administration server (DAS) user name: bpmadmin

If the fenced user or administration (DAS) server user name exists, clear the **New User** check box. Click **Re-Validate** to confirm that your choices are correct.

Restriction: User names must not contain National Language Strings (NLS).

14. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
15. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 [Installation Manager updates](#)

Installing IBM Business Process Manager silently:

You can install the IBM Business Process Manager product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related tasks:

“Preparing Linux systems for installation” on page 135

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing silently using the command line:

You can install IBM Business Process Manager silently using the command line. With this method, you only need to run one command to perform the installation.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the ejbdeploy and thinclient features.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

```
extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.

3. Run the following command:

```
extract_directory/IM/tools/imcl install list_of_product_IDs -acceptLicense  
-installationDirectory location -repositories repository  
-properties key=value,key=value -showVerboseProgress -log logName.log
```

where:

- *list_of_product_IDs* is a list of the IDs for the products and features that you want to install. You must include the required features. The syntax is *productID,feature,feature*, with multiple products separated by spaces.

Table 66. Product IDs

Product	Product ID	Feature and Description
IBM BPM Advanced	com.ibm.bpm.ADV.V80	<ul style="list-style-type: none"> • bpmAdv.nonprod: Test, staging, or development use. You must specify either this feature or bpmAdv.prod. • bpmAdv.prod: Production use. You must specify either this feature or bpmAdv.nonprod. • wps.client.feature: Required. Run client applications that interact with a IBM BPM Advanced: Process Server in the same cell. • wps.server.feature: Required. Full installation of IBM BPM Advanced: Process Server, which includes the client code and the server code.

Table 66. Product IDs (continued)

Product	Product ID	Feature and Description
WebSphere Application Server Network Deployment	com.ibm.websphere.ND.v80	<ul style="list-style-type: none"> core.feature: Required. WebSphere Application Server core content. ejbdeploy: Required. Pre-Enterprise JavaBeans (EJB) 3.0 modules. thinclient: Required. Stand-alone thin clients and resource adapters. embeddablecontainer: Embeddable EJB container. samples: Sample applications feature. com.ibm.sdk.6_32bit: 32-bit software development kit (SDK). You must specify either this feature or com.ibm.sdk.6_64bit. com.ibm.sdk.6_64bit: 64-bit SDK. This feature can only be selected on a 64-bit system. You must specify either this feature or com.ibm.sdk.6_32bit.
Installation Manager	com.ibm.cic.agent	<ul style="list-style-type: none"> agent_core: Installation Manager core content. agent_jre: Installation Manager Java Runtime Environment (JRE).
DB2 Express for Linux 32-bit	com.ibm.ws.DB2EXP97.linuxia32	n/a: DB2 must match the operating system and bitness.
DB2 Express for Linux 64-bit	com.ibm.ws.DB2EXP97.linuxia64	n/a: DB2 must match the operating system and bitness.

- *location* is the path to the directory where you want to install the products.
- *repository* is the path to the repository where you have extracted the files, one of the following directories:
`extract_directory/repository/repos_32bit`
`extract_directory/repository/repos_64bit`

For more than one repository, separate the repository locations with commas.

- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas. Create encrypted passwords using the IBM Installation Manager.

Table 67. Keys

Key	Description
user.db2.use.existing	<p>If you are using an existing DB2 setup, add the following line exactly: <code>user.db2.use.existing=true</code></p> <p>If DB2 must be installed, specify the value <i>false</i>.</p>
user.db2.port	Port for the DB2 database. The default value is 50000.
user.db2.instance.username	DB2 instance user name. The default value is bpminst.
user.db2.instance.password	Encrypted password for the DB2 instance user name.

Table 67. Keys (continued)

Key	Description
user.db2.fenced.newuser	The value <i>true</i> is for a new user. The value <i>false</i> is for an existing user.
user.db2.fenced.username	Fenced user name. The default value is bpmfenc.
user.db2.fenced.password	Encrypted Password for the fenced user name.
user.db2.das.newuser	The value <i>true</i> is for a new user. The value <i>false</i> is for an existing user.
user.db2.das.username	Administration server (DAS) user name. The default value is bpmadmin.
user.db2.das.password	Encrypted password for the administration server user name.

- *logName* is the name of the log file to record messages and results.

Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for `imcl`.

Installation Manager installs the list of products and writes a log file to the directory that you specified. The log file is empty if there are no errors or warnings.



The following example installs IBM Business Process Manager Advanced, WebSphere Application Server Network Deployment, and DB2 Express on Linux.

```
imcl install com.ibm.bpm.ADV.V80,bpmAdv.prod,wps.client.feature,wps.server.feature
com.ibm.websphere.ND.v80,core.feature,ejbdeploy,thinclient,embeddablecontainer,samples,com.ibm.sdk.6_32bit
com.ibm.ws.DB2EXP97.linuxia32 -acceptLicense
-installationDirectory /usr/IBM/BPM80 -repositories /usr/tmp/BPM/repository
-properties user.db2.instance.username=bpmadmin,user.db2.instance.password=Vvrs88V/a9BUdxwodz0nUg==
-showVerboseProgress -log silentinstall.log
```

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the `manageprofiles` command-line utility. Only profiles created with the Profile Management Tool or `manageprofiles` can be used in production.

Related reference:

-  [Command-line arguments for imcl](#)
-  [IBM Business Process Manager Advanced system requirements](#)

Related information:

-  [IBM WebSphere Application Server Information Center](#)

Installing silently using a response file:

You can install IBM Business Process Manager Advanced silently by creating a response file and then running a command to use that response file to install the product.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager

- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

By using response files, you can simplify the silent installation and reduce error in the process because you set up your installation options once in a saved, sharable file that can be used on one or more machines. The installation software provides sample response files for each supported operating system and bit version. You can use an unmodified sample response file to perform a silent installation using default settings, or you can modify the response file to set particular values. The comments in the sample response files provide detailed instructions and information about setting the values.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

```
extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Create the response file that will install the required base products and IBM Business Process Manager Advanced. Copy a sample response file, suitable for your bit version and user access level, from the following directory:

```
extract_directory/responsefiles/BPM/
```

Alternatively, you can create a response file by recording your actions in Installation Manager. When you record a response file, the selections that you make in Installation Manager are stored in an XML file. When you run Installation Manager in silent mode, Installation Manager uses the data in the XML response file to perform the installation.

3. The default values provided in the sample response files will perform a basic install, but you should review the file and its comments, and modify the parameters as needed for your environment and the access level of your user ID.

In particular, review the following parameters and values:

- For non-root user IDs, check that all location variables point to locations that the user has permissions for.
 - Repository location. If you are not running directly from the *extract_directory*/responsefiles/BPM/ directory, point to the location of your installation repository. The repository can be local or remote. If you are running from the DVD, copy the response file from the DVD and point it back at the repository on the DVD.
 - Installation location (**installLocation**) where IBM Installation Manager is already installed or will be installed.
 - Installation location where IBM Business Process Manager Advanced will be installed.
 - Eclipse location (**eclipseLocation**).
 - The list of features for the product.
 - Production or non-production use.
 - If you are installing DB2 Express, follow the instructions in the response file for providing the necessary user IDs and passwords. Use the instructions in step 1 for generating the encrypted passwords to include.
4. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.
 5. Run the following command:

Root user:

```
extract_directory/IM/installc -acceptLicense input
  extract_directory/responsefiles/BPM/response_file_name.xml
  -log preferred_log_location/silent_install.log
```

Nonroot user:

```
extract_directory/IM/userinstc -acceptLicense input
  extract_directory/responsefiles/BPM/response_file_name.xml
  -log preferred_log_location/silent_install.log
```

Installation Manager installs any required prerequisites and IBM Business Process Manager Advanced, and writes a log file to the directory you specified.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

Related tasks:

 Installing silently by using Installation Manager

 Recording a response file with Installation Manager

Related reference:

 IBM Business Process Manager Advanced system requirements

Granting write permission of files and directories to nonroot users for profile creation or augmentation:

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root/profiles* directory of the product.

Restrictions:

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the `profilers` group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
 - a. Create a group named **profilers**, which will contain all users who can create profiles.
 - b. Create a user named **user1**, who can create profiles.
 - c. Add users **product_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
 - Create the *install_root/logs/manageprofiles* directory:
`mkdir install_root/logs/manageprofiles`
 - Create the *install_root/properties/fsdb* directory:
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the `profileRegistry.xml` file as the installer. For this example, the file path is:
`install_root/properties/profileRegistry.xml`

Add the following information to the `profileRegistry.xml` file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable `$WASHOME` is the IBM Business Process Manager root installation directory `/opt/IBM/WebSphere/AppServer`.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

The root user and the nonroot user can use the same tasks to manage profiles.

Configuring profiles and databases:

After you install the product, if you did not use typical installation, you must create or augment one or more profiles to define the runtime environment. Before starting a profile, you must have configured the databases that are to be used with the profile.

Configuring a stand-alone environment with a DB2 database server on Linux:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 database server.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database™, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the /etc/profile file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root user to the /etc/group administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

Creating database design files for DB2:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/uti/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/uti/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
```

- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2) bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.
 [info] Completed database components can be edited to change existing or defaulted property values.
 [info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI_BPC : [status = not complete]
- (3) [BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4) [BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI_BSPACE : [status = not complete]
- (6) [SibME] WBI_BPC_ME : [status = not complete]
- (7) [SibME] WBI_CEI_ME : [status = not complete]
- (8) [SibME] WBI_SCA_APP_ME : [status = not complete]
- (9) [SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10) [SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11) [SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI_CommonDB is not complete with 1 remaining item(s):
 [1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool

is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]
```

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.standalone is complete with 0 remaining item(s):
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```


12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
After the last set of scripts is generated, you see information similar to the following example:
[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.

- The location of the JDBC driver on the system where the server profile will be created.
- The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/Util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/Util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :
```

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm_performancedw
- (4) bpm_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 1 to select **(1) DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Database User name[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1) false
- (2) true

Please enter the number for the Is this database for a Process Center? [default=false] :
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=TWSSPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=TWSTEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database server host[default=] :  
Database server port[default=50000] :  
Data source user name[default=] :  
Data source password[default=] :  
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with a DB2 database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the DB2 database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

If you plan to configure the databases on a remote server, you must have created the databases before beginning to create or augment the profile. You can create databases on the local server or use existing databases on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103:db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

Creating Process Center stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the *bin* directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including *tw_admin* and *tw_user*. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower® signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.

- b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 68. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/db2/java.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

- 18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
- 19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
- 20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

- 21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
- 22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.

5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Advanced:** If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.
If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.
You must specify a user name under which the service runs.
To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.
14. **Optional: Advanced:** To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 69. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
 - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response

- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWdbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:

```

```

# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumber where profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName

```

```

#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod

```

```

#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile

```



```

#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:

```

```

# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true

```

```

#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:

```

```

# The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
# The schema name for the Common database. If no value is specified, the
# following default database schema names are used:
# DB2 database
# The default schema name is the value specified by the dbUserId parameter.
# The -cdbSchemaName parameter is valid for all database types except for
# Oracle and Microsoft SQL Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```



```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

```

```

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.

```

```

#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter:  webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,

```

```

# the IBM Forms Server install root might be
# C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
# Parameter:  configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false
#####
# Parameter:  configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#####
configureBPC=true

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM

```

```

# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a

```

```

# trailing number, for example:
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile

```

```

# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter

```



```

# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.

```

```

# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem

```

```

#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password

```

```

#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: dbHostName
#
# Description:
# The database server host name or IP address. The default value is the
# local host name. Required when the -bpmdbDesign parameter was not
# specified. (It is not required if the -bpmdbDesign parameter is
# specified.)
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbServerPort
#
# Description:
# The database server port number. Depending on the database you are using,
# you can specify a different port number instead of the default port
# number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdriver/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB

```

```

#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

```



```

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true

```

```
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
```

```

# path for the parameter.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
# Set a performance tuning level appropriate for the profile you are
# creating.
# This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
# development
# standard
# production
# Default Values:
# None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
# shortHostNameCellCellNumber

```

```

#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the

```



```

# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:

```

```

# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####

```

```

# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#

```

```

# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter:  dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:

```

```

# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:

```

```

# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid

```



```

#dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####

```

environmentType=Test

```
#####  
# Parameter: isDeveloperServer  
#  
# Description:  
#   Specifies whether the server is intended for development purposes only.  
#   This parameter is useful when creating profiles to test applications on a  
#   non-production server prior to deploying the applications on their  
#   production application servers. If -isDeveloperServer is set when  
#   creating an IBM Business Process Manager profile, then a preconfigured VMM  
#   file repository is installed. This file repository contains a sample  
#   organization that can be used to test Business Process Choreographer  
#   people resolution, ready for you to use as is.  
#  
# Valid Values:  
#   false  
#   true  
# Default Values:  
#   false  
#####  
#isDeveloperServer=false
```

```
#####  
# Parameter: webFormConfig  
#  
# Description:  
#   Indicates if Business Space is configured to use IBM Forms Server to work  
#   with Human Task Management widgets. The default value for this parameter  
#   is false. Indicate true to configure Business Space to use IBM Forms  
#   Server. Both the webFormConfig and webFormInstallRoot parameters are  
#   required to configure IBM Forms Server. This parameter is valid for  
#   stand-alone server profiles only.  
#   Note: IBM Forms Server configuration using these parameters is only valid  
#   for local IBM Forms Server installations.  
#
```

```
# Valid Values:  
#   false  
#   true  
# Default Values:  
#   false  
#
```

```
# Parameter: webFormHTTP  
#
```

```
# Description:  
#   Specifies the location of the IBM Forms Server Translator. The default URL  
#   for the location is http://localhost:8085/translator. No. This parameter  
#   is valid only if the -webFormConfig parameter is set to true. It is valid  
#   for stand-alone server profiles only.  
#
```

```
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#
```

```
# Parameter: webFormInstallRoot  
#
```

```
# Description:  
#   Specifies the full path where IBM Forms Server is installed. For example,  
#   the IBM Forms Server install root might be  
#   C:/IBM/IBMFoMServer/4.0/WebFormServer. No. This parameter is valid only  
#   if the -webFormConfig parameter is set to true. It is valid for  
#   stand-alone server profiles only.  
#
```

```
# Valid Values:  
#   Varies  
# Default Values:
```

```

# None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#####
configureBPC=true

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.

```

```

#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the

```

```

#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber

```

```

#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####

```



```

# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this

```

```

# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An

```

```

# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for

```

```

# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbHostName
#
# Description:
# The database server host name or IP address. The default value is the
# local host name. Required when the -bpmdbDesign parameter was not
# specified. (It is not required if the -bpmdbDesign parameter is
# specified.)
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbServerPort
#
# Description:
# The database server port number. Depending on the database you are using,
# you can specify a different port number instead of the default port
# number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None

```

```

#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database.  The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse.  The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name.  Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:

```



```

# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName

```

```

#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
# Parameter:  environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify

```

```

# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.
 The **Typical** option augments a profile with default configuration settings.
 The **Advanced** option lets you specify your own configuration values for a profile.
- Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:
- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including *tw_admin* and *tw_user*. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 70. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.

13. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.

- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 71. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.

Table 71. Required database configuration fields for DB2 Database (continued)

Field	Action needed
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

15. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
16. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
 - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response

- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring DB2 databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/ProcessServer/DB2/ at profile creation time.
2. Run the following script.

```
configProcessServerDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/PerformanceDW/DB2/ at profile creation time.
2. Run the following script.

```
configPerformanceDWDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
 - Chosen not to run the database scripts to initialize the Common database during the configuration.
 - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/DB2/*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

configCommonDB.sh createDB - creates the database and tables

configCommonDB.sh - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**

- g. `createTable_governancerepository.sql`
- h. `createTable_lockmanager.sql`
- i. `createTable_mediation.sql`
- j. `createTable_Recovery.sql`
- k. `createTable_RelationshipService.sql`
- l. `insertTable_CommonDB.sql`

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is `install_root/profile/dbscripts/BusinessSpace/node_name_server_name/DB2/CMNDB`.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:
configBusinessSpaceDB.sh Connect to the database, bind the Command Line Interface to the Business Space database using the **db2 bind DB2_installation_directory\bnd\@db2cli.lst blocking all grant public** command, and then reset the database connection.

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.timestamp.log`. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Configuring a stand-alone environment with a DB2 for z/OS database server on Linux:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

Creating database design files and database scripts (Linux stand-alone):

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS:

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help  
    display help info.
```

```
-e db_design_file_name  
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```

-v db_design_file | db_scripts_output_directory
when a db_design_file is given, validation will be done on the specified database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-z0S/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-z0S/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-z0S/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the `DbDesignGenerator.sh` command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- (1)Create a database design for Standalone profile or Deployment Environment
- (2)Create a database design for a single component
- (3)Edit an existing database design
- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI_BPC : [status = not complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = not complete]
- (6)[SibME] WBI_BPC_ME : [status = not complete]
- (7)[SibME] WBI_CEI_ME : [status = not complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = not complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):  
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, `WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB`. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed  
(2)DB2-zOS  
(3)Oracle  
(4)SQL Server
```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.  
Database name[default=CMNDB] :  
Data source user name[default=] :  
Database schema name (SQLID)[default=] :  
Bufferpool of 4K size for tables[default=BP1] :  
Bufferpool for indexes[default=BP2] :  
Bufferpool of 4K size for LOB data[default=BP3] :  
Bufferpool of 8K size for tables[default=BP8K1] :  
Bufferpool of 16K size for tables[default=BP16K1] :  
Bufferpool of 32K size for tables[default=BP32K1] :  
Storage group name[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter `'s'`; or enter anything else to continue :

9. To configure the data source properties component, type anything other than *s* and press Enter. To skip this configuration and accept the defaults, type *s* and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

Note: For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

```
[info] Please pick one of the following [Is this database for a Process Center?(s)] :
```

- ```
(1)false
(2>true
```

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.standalone**, after you configure the master component, you must manually configure the **BPM\_PerformanceDW** and **BPM\_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

```
[info] Please pick one of the following [database component(s)] :
```

- ```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = complete]
(2)[BPC]               WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSpace]           WBI_BSPACE : [status = complete]
(6)[SibME]            WBI_BPC_ME : [status = complete]
(7)[SibME]            WBI_CEI_ME : [status = complete]
(8)[SibME]            WBI_SCA_APP_ME : [status = complete]
(9)[SibME]            WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe]           BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe]           BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

```
Please enter the number for the database component :12
```

```
[status] bpm.advanced.standalone.topology is not complete with 6 remaining item(s):
[ 1 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[ 2 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'catalog' for VCAT is empty.
```



```
[ 3 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminUserName' for SECURITY_ADMIN_USER is empty.
[ 4 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
[ 5 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[ 6 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'catalog' for VCAT is empty.
```

- When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

- Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

- To generate the database scripts for each component in the directory that you specified earlier, type **y** and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment:

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
```

```
Database name[default=BPMDDB] :  
Data source user name[default=] :  
Database schema name (SQLID)[default=] :
```

```
[info] Please pick one of the following [Is this database for a Process Center?(s)] :
```

```
(1>false  
(2>true
```

```
Please enter the number for the Is this database for a Process Center? [default=false] :
```

```
The user ID you use for administrative security[default=] :
```

```
The password for the name specified with the adminUserName parameter[default=] :
```

```
Bufferpool of 4K size for tables[default=BP1] :
```

```
Bufferpool of 4K size for LOB data[default=BP3] :
```

```
Bufferpool of 8K size for tables[default=BP8K1] :
```

```
Bufferpool of 16K size for tables[default=BP16K1] :
```

```
Bufferpool of 32K size for tables[default=BP32K1] :
```

```
Storage group name[default=] :
```

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

```
To skip data source properties, enter 's'; or enter anything else to continue :
```

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
```

```
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database Location on z/OS system[default=] :
```

```
Database server host[default=] :
```

```
Database server port[default=] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. To generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/dbscripts/DB2-zOS/ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Related tasks:

“Creating the messaging engine data stores” on page 357

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Generating scripts for creating database objects in DB2 for z/OS:

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
 - Database server details

- Data source details
- Location of the JDBC driver on the system where the server profile will be created
- User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z®. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory `/tmp/DB2-zOS/W1MYDB`. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as `/tmp/DB2-zOS/W8CELLDB`. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
 - To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as `/tmp/DB2-zOS/W8S1DB`. You can later run the generated schema to create the database objects for these components in the W8S1DB database.
1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.

2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: `cd /opt/IBM/WebSphere/AppServer/util/dbUtils`

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db_design_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

Tip: In the database design file, the directories for the SQL scripts are defined by the *ddl_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

Note: The **createDB2.sh** script requires execute permissions.

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```


Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with a DB2 for z/OS database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Creating Process Center stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next** .
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating Process Server stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default

language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones.

If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this

Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

16. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
 - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
 - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2zOS.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2zOS.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2zOS.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2zOS.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2zOS.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2zOS.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2zOS.response`

- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWdbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
```

```

# path for the parameter.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
# Set a performance tuning level appropriate for the profile you are
# creating.
# This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
# development
# standard
# production
# Default Values:
# None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (Dmgr)
# shortHostNameCellCellNumber

```

```

#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:

```

```

# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.

```

```

# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true, the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies

```



```

# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:

```

```

# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
# Database name for Process Server database. The value specified for
# -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrUserId
#
# Description:
# The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrPassword
#
# Description:
# The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#

```

```

# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.

```

```

# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: webFormInstallRoot
#
# Description:
# Specifies the full path where IBM Forms Server is installed. For example,
# the IBM Forms Server install root might be
# C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
# Parameter: configureBRM
#
# Description:
# Configures the business process rules manager. The default value is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
configureBRM=false

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT
#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2z0S.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2z0S.response

```



```

#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####

```

```

profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#

```

```

# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name

```

```

#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:

```

```

# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

```

```

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```



```

# CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

```

```

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```

#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#

```



```

# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service

```

```

# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbdbDesign that was used in previous

```

```

# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true
#####

```

```

# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter:  dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer

```



```

#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter
#     is false. Indicate true to configure Business Space to use IBM Forms
#     Server. Both the webFormConfig and webFormInstallRoot parameters are
#     required to configure IBM Forms Server. This parameter is valid for
#     stand-alone server profiles only.
#     Note: IBM Forms Server configuration using these parameters is only valid
#     for local IBM Forms Server installations.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter: webFormHTTP
#
# Description:
#     Specifies the location of the IBM Forms Server Translator. The default URL
#     for the location is http://localhost:8085/translator. No. This parameter
#     is valid only if the -webFormConfig parameter is set to true. It is valid
#     for stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: webFormInstallRoot
#
# Description:
#     Specifies the full path where IBM Forms Server is installed. For example,
#     the IBM Forms Server install root might be
#     C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#     if the -webFormConfig parameter is set to true. It is valid for
#     stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#     Configures the business process rules manager. The default value is false.
#
# Valid Values:

```

```

# false
# true
# Default Values:
# false
#####
configureBRM=false

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.

```

```

#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:

```

```

#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process

```

```

# Center). It is required for a stand-alone process server (standard or
# advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.

```

```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName

```



```

#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```

```

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:

```

```

# false
# true
# Default Values:
# false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
# Database name for Process Server database. The value specified for
# -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None

```

```

#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#

```

```

# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
 9. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 10. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 11. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
 12. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Augmenting stand-alone profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```


For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.

- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
10. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
11. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
12. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
13. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - *default.procctr*: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - *default.procctr.adv*: for a Process Center stand-alone profile for IBM BPM Advanced.
 - *default.procsvr*: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - *default.procsvr.adv*: for a Process Server stand-alone profile for IBM BPM Advanced.
 - *default.esbserver*: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the *augment* parameter to make changes to an existing profile with an augmentation template. The *augment* parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- *PC_Std_DMGr_DB2.response*
- *PC_Std_DMGr_DB2z0S.response*
- *PC_Std_DMGr_Oracle.response*
- *PC_Std_DMGr_SQLServer.response*
- *PC_Adv_DMGr_DB2.response*
- *PC_Adv_DMGr_DB2z0S.response*
- *PC_Adv_DMGr_Oracle.response*
- *PC_Adv_DMGr_SQLServer.response*
- *PC_Std_Managed_DB2.response*
- *PC_Std_Managed_DB2z0S.response*
- *PC_Std_Managed_Oracle.response*
- *PC_Std_Managed_SQLServer.response*
- *PC_Adv_Managed_DB2.response*
- *PC_Adv_Managed_DB2z0S.response*
- *PC_Adv_Managed_Oracle.response*
- *PC_Adv_Managed_SQLServer.response*
- *PC_Std_StandAlone_DB2.response*
- *PC_Std_StandAlone_DB2z0S.response*

- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
 -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating and configuring DB2 for z/OS databases after stand-alone profile creation:

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

DB2 for z/OS authorization prerequisites:

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as `#DB_USER#`. These permissions are available in the `createDatabase.sql` database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the `#DB_USER#` user to create sequences and stored procedures with a schema qualifier of `#SCHEMA#`:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database `#DB_NAME#` can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF®) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:
 - Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```

GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;

```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named S1CELL, and connect user WSADMIN to it, as follows:

```

INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
WSADMIN CONNECT 000000 NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE RESUME DATE=NONE

```

Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating databases in the DB2 for z/OS subsystem:

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

Considerations for choosing the createDB2.sh script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two `.sql` files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

Related tasks:

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Configuring the DB2 command line processor:

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
 - Modify the CLASSPATH environment variable to include the clp.jar file.
 - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
 - Define the **db2** command as an alias for the command that starts the command line processor.
 - Specify the DB2JccConfiguration.properties file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the .profile or /etc/profile file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=
/file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=
/wasv8config/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:
 - A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
 - A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

Tip: When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSN'TIPP ISPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script” on page 352

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 354. When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

Creating DB2 for z/OS database objects using the createDB2.sh script:

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCreate  
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

-RunSQL

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP  
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

Tip: The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 350

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using the DB2 command line processor:

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate  
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

Tip: By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two `.sql` files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

The `z_schema.sql` and `z_schemaProc.sql` files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The `z_schemaProc.sql` file contains SQL statements for stored procedures, and is generated in addition to `z_schema.sql` because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the `z_schema.sql` file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the `z_schemaProc.sql` file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 350

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

Tip: The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f IS08859-1 createTable_AppScheduler.sql >  
createTable_AppScheduler_EBCDIC.sql
```

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS™ data sets.

Tip: If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t IS08859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >  
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

@STOGRP@ = DB2 Storage group name
 @DB_NAME@ = DB2 Database name
 @SCHEMA@ = DB2 SQLID (Schema Qualifier)
 @BP4K@ = Buffer Pool of 4k Size
 @BPINDEX@ = Buffer Pool Index
 @VOLUMES@ = DB2 comma separated list of volume-ids
 @VCAT@ = DB2 integrated catalog facility name
 @DB_USER@ = The user ID which will be used to create the database objects

4. To create database objects outside of the z/OS UNIX environment by using SPUIFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

Tip: If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (`z_schema.sql` and `z_schemaProc.sql`), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

Related tasks:

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 354

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

Creating the messaging engine data stores:

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
- When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
 - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
 - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.

1. For each messaging engine, complete the following steps to generate the required database scripts:
 - a. Run the database design tool (**DbDesignGenerator**) from the `install_root/util/dbUtils` directory.
 - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
 - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

Tip: As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

Related tasks:

“Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment” on page 279

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS” on page 273

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

Note: Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Setting the correct schema name for the messaging engines:

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
 - a. Select **Messaging engines**, and then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Verify that the messaging engines can be started:
 - a. Log out of the administrative console.
 - b. Stop, and then restart the server.

- c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"  
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

Configuring a stand-alone environment with an Oracle database server on Linux:

You can configure a stand-alone environment for IBM Business Process Manager using an Oracle database server.

Setting up Oracle databases:

You or your database administrator (DBA) can configure the Process Server database, Performance Data Warehouse database, and Common database before, after, or during creating or augmenting a profile.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

You have the following options for creating and configuring the databases:

- You or your DBA creates the databases in advance and the Profile Management Tool configures them for you during profile creation or augmentation.
- You or your DBA creates a database design file using the database design tool, and you pass the file to the Profile Management Tool or **manageprofiles** command-line utility during profile creation or augmentation.
- You postpone the database configuration, and you or your DBA creates the required databases and database tables later by running the scripts that are generated during profile creation or augmentation.

Creating database design files for Oracle:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with Oracle:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/uti/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/uti/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSpace]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):  
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```
(1)DB2-distributed  
(2)DB2-zOS  
(3)Oracle  
(4)SQL Server
```

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.  
Database name(SID) [default=CMNDB] :  
Database User name[default=] :  
Database schema [default=] :  
System user name(this is required ONLY for creating the database as a part of standalone profile  
creation.)[default=] :  
System password(this is required ONLY for creating the database as a part of standalone profile  
creation.)[default=] :  
Database Location(this is required ONLY for creating the database as a part of standalone profile  
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than *s* and press Enter. To skip this configuration and accept the defaults, type *s* and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database server host[default=] :  
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8  
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :  
Data source user name[default=] :  
Data source password[default=] :  
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle  
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----
```

```
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components,  
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]  
(2)[BPC] WBI_BPC : [status = complete]  
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]  
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]  
(5)[BSPACE] WBI_BSPACE : [status = complete]  
(6)[SibME] WBI_BPC_ME : [status = complete]  
(7)[SibME] WBI_CEI_ME : [status = complete]  
(8)[SibME] WBI_SCA_APP_ME : [status = complete]  
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]  
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]  
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]  
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.standalone**, you must manually configure the **BPM_PerformanceDW**, **BPM_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :3

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.

Edit this database component? (y/n) [default=y] :

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB
After the last set of scripts is generated, you see information similar to the following example:
[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with Oracle in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.

- The location of the JDBC driver on the system where the server profile will be created.
- The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/Util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/Util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :
```

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm_performancedw
- (4) bpm_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 3 to select **(3) Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=BPMDDB] :

Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1) false
- (2) true

Please enter the number for the Is this database for a Process Center? [default=false] :
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database server host[default=] :  
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8  
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :  
Data source user name[default=] :  
Data source password[default=] :  
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle  
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```


9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with an Oracle database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the Oracle database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

To configure the database information while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M  
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];  
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE  
[tablespace_name];  
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE  
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;  
SQL> GRANT ALL PRIVILEGES TO PDWDBA;  
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;  
SQL> GRANT execute on DBMS_LOCK to PDWDBA;  
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating Process Center stand-alone profiles with an Oracle database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 72. Required database configuration fields for Oracle

Field	Action needed
Common database	For the Common database, enter values for the following parameters: User name Enter a user name for the Common database (such as CMNDBA). Password Enter the password that you specified when you created the user in Oracle. Confirm password Confirm the password by reentering it.

Table 72. Required database configuration fields for Oracle (continued)

Field	Action needed
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
- When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
- The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with an Oracle database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.

- Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including *tw_admin* and *tw_user*.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
 10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. **Advanced:** On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy

server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 73. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.

2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response

- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```

#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:

```

```

# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:

```



```

# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#

```

```

# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####

```

```

# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter:  dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrMePassword

```

```

#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine. This
#   parameter is required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeUserId
#
# Description:
# The default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
# password during profile creation and if you specified ORACLE for dbType,
# this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbAppMePassword
#
# Description:
# The password for the user id of the dbAppMeUserId parameter. If you enter
# a user-specified a user name and password during profile creation and if
# you specified ORACLE for dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
# The default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
# password during profile creation and if you specified ORACLE for dbType,
# this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCeiMePassword
#
# Description:
# The password for the user id of the dbCeiMeUserId parameter. If you enter
# a user-specified a user name and password during profile creation and if
# you specified ORACLE for -dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
#   user-specified a user name and password during profile creation, this
#   parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword. If you enter a user-specified a user name and password during
#   profile creation, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test

```



```

# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:

```

```

# Varies
# Default Values:
# None
#
# Parameter: webFormInstallRoot
#
# Description:
# Specifies the full path where IBM Forms Server is installed. For example,
# the IBM Forms Server install root might be
# C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
# Configures the business process rules manager. The default value is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
# Determines whether the Business Process Choreographer sample configuration
# is created.
# The following conditions apply:
# Only DB2 databases are supported.
# The default is set to true for DB2.
# It uses its own schema within the common database (CMNDB) and the schema
# name is always BPEDB.
# Business Process Choreographer database parameters set in the Database
# Design Tool and generated to the database design file are ignored by
# Business Process Choreographer at profile creation time.
# See -bpmdbDesign for information about the database design file.
# Note: If these conditions are too restrictive, consider setting
# -configureBPC to false and using other documented methods to configure
# Business Process Choreographer.
#
# Valid Values:
# false
# Default Values:
# false
#####
configureBPC=false

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
```

```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is

```



```

#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:

```

```

# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For

```

```

#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter: dbLocation
#
# Description:
#     The directory of the ORACLE_HOME directory. This parameter is required
#     when the parameter dbDelayConfig is set to false. No, this parameter is
#     only valid if you are using Oracle databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
dbLocation=db_location
#####
# Parameter: dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle

```

```

#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     orcl
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID. Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database. The value specified for
#     -procSvrDbName and -perfDWDbName can be the same.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None

```

```

#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine. This
#   parameter is required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserID
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

```

```

#####
# Parameter:  environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
isDeveloperServer=false

```


Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```

#####
templatePath=BPM/default.procsvr.adv
#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01
#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard
#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter: processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#

```

```

# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service

```

```

# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbdbDesign that was used in previous

```



```

# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

```

```

#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName can be the same.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine. This
# parameter is required.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbAppMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd
#####
# Parameter: dbCeiMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd
#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
#   user-specified a user name and password during profile creation, this
#   parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: dbSysMePassword
#
# Description:
# The default password for the SCA system bus messaging engine is
# dbPassword. If you enter a user-specified a user name and password during
# profile creation, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#

```



```

# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

```

```
#####
# Parameter:  configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter:  configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
configureBPC=false
```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
```

```

# Standalone PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be

```

```

# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword

```

```

# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#

```



```

# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winServiceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winServiceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winServicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winServiceCheck
#
# Description:

```

```

# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false

```

```

#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####

```

```

dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

```

```

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####

```

```

perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine. This
#   parameter is required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:

```



```

# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with an Oracle database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including *tw_admin* and *tw_user*. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.

- You must input the database service name for each database.
 - The database must already exist.
- c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
- d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 74. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>

Table 74. Required database configuration fields for Oracle (continued)

Field	Action needed
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 13. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
 14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with an Oracle database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 75. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
 15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
 - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_Standalone_DB2.response
- PC_Std_Standalone_DB2zOS.response
- PC_Std_Standalone_Oracle.response
- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response

- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the

bootstrapProcessServerData command before you try to start or use Process Server or Process Center.

Configuring Oracle databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/ProcessServer/Oracle/ at profile creation time.

2. Run the following script.

```
configProcessServerDB.sh
```

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/PerformanceDW/Oracle/ at profile creation time.

2. Run the following script.

```
configPerformanceDWDB.sh
```

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
 - Chosen not to run the database scripts to initialize the Common database during the configuration.
 - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/Oracle/*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run the following command:

configCommonDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install_root/profile/dbscripts/BusinessSpace/node_name_server_name/Oracle/CMNDB*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:

configBusinessSpaceDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:

- a. **createDatabase.sql**
- b. **createTable_BusinessSpace.sql**
- c. **createGrant_BusinessSpace.sql**

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Configuring a stand-alone environment with an SQL Server database server on Linux:

You can configure a stand-alone environment for IBM Business Process Manager using a Microsoft SQL Server database server.

Creating SQL Server databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardi software.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default

schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
 - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
 - c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
 - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.

- c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the x64 folder.
- d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

Creating database design files for SQL Server:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with SQL Server:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
  display help info.
```

```

-e db_design_file_name
  edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
  when a db_design_file is given, validation will be done on the specified
  database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
  generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.

```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]

```

Please enter the number for the design option :

- To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

- To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

- Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

- To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

- Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :

Data source password[default=] :

MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the **BPM_ProcessServer**, you must manually configure the **BPM_PerformanceDW** component for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(5)[save and exit]
```

```
Please enter the number for the database component :2
```

```
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with SQL Server in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/uti/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/uti/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm_performancedw
- (4) bpm_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 4 to select **(4) SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.
Database name[default=BPMDDB] :
Database server host[default=] :
Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1)false
(2>true

Please enter the number for the Is this database for a Process Center? [default=false] :
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

(1) false
(2) true

Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with an SQL Server database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the SQL Server database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Restriction: The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 76.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner dbo using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDBE00 (Business Process Choreographer Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)

- MEDPE00 (Performance Data Warehouse messaging engine)

IBM Business Process Manager packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the Detailed hardware and software requirements for IBM Business Process Manager page.

Note: You are responsible for providing JDBC driver levels outside of what is packaged with IBM Business Process Manager.

You must configure XA transactions after the database is installed and before you start the server.

Creating Process Center stand-alone profiles with an SQL Server database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.

5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.
14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.
 - e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step
16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 77. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 78. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with an SQL Server database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.

2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.

- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 79. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.

Table 79. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 80. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
 - **Windows Authentication:** With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
19. **Advanced:** On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 20. **Advanced:** Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
 - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
 - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`

- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a stand-alone profile that includes the CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space components in one database.

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates/BPM/  
default.procsvr.adv -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath  
path to JDBC driver -dbUserId userID -dbPassword password -dbCommonForME true -dbCeiMeSchemaName  
ceiSchema -dbSysMeSchemaName sysSchema  
-dbAppMeSchemaName appSchema
```

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT  
# *****  
#  
# Licensed Materials - Property of IBM  
# 5725-C94  
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.  
# This sample program is provided AS IS and may be used, executed, copied  
# and modified without royalty payment by customer (a) for its own  
# instruction and study, (b) in order to develop applications designed to  
# run with an IBM WebSphere product, either for customer's own internal  
# use or for redistribution by customer, as part of such an application,  
# in customer's own products.  
#  
# *****  
# END COPYRIGHT  
  
#####  
# These response file has the applicable parameters for creating a  
# Standalone PC Adv profile that uses SQLServer.  
#  
# Depending on your environment, you may need to change the default values.  
#  
# To create a profile with this response file specify:  
#  
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_SQLServer.response  
#  
# If you use the -response parameter, it must be the only parameter on  
# the command-line. If you include any manageprofile parameters  
# (in addition to -response) on the command-line, the response file  
# is ignored. And default WebSphere Application server profile will be created .  
#####  
  
#####
```



```

# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.

```

```

# This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
# development
# standard
# production
# Default Values:
# None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: serverName
#
# Description:
# Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####

```

```

# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this

```

```

# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An

```

```

# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter:  dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for

```

```

# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# MSSQLSERVER_MICROSOFT
# Default Values:
# MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#

```



```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword

```

```

#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_ME_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeSchemaName

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd
#####
# Parameter: dbCeiMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeSchemaName=sys_me_schemaName
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter

```

```

#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality. Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers. If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#####
isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter

```

```

# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: webFormInstallRoot
#
# Description:
# Specifies the full path where IBM Forms Server is installed. For example,
# the IBM Forms Server install root might be
# C:/IBM/IBMFoMServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
# Configures the business process rules manager. The default value is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
# Determines whether the Business Process Choreographer sample configuration
# is created.
# The following conditions apply:
# Only DB2 databases are supported.
# The default is set to true for DB2.

```



```

# It uses its own schema within the common database (CMNDB) and the schema
# name is always BPEDB.
# Business Process Choreographer database parameters set in the Database
# Design Tool and generated to the database design file are ignored by
# Business Process Choreographer at profile creation time.
# See -bpmdbDesign for information about the database design file.
# Note: If these conditions are too restrictive, consider setting
# -configureBPC to false and using other documented methods to configure
# Business Process Choreographer.
#
# Valid Values:
# false
# Default Values:
# false
#####
configureBPC=false

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####

```

```

# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####

```

```

#applyPerfTuningSetting=standard
#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:

```

```

# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true

```

```

# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this

```

```

#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbldbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
#bpmdbDesign=design_file

#####
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
#####
dbCreateNew=false

```



```

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#

```

```

# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#

```

```

# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database. For Oracle, the
# default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
# Process Manager profiles and advanced deployment manager (dmgr.*.adv)
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
# and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter. Required for
# all stand-alone IBM Business Process Manager profiles and advanced
# deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
# to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbWinAuth
#
# Description:
# An optional parameter that is valid in all default profile templates.
# Specify true to indicate that you are running SQL Server using integrated
# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

# false
# true
# Default Values:
# None
#####
#dbWinAuth=true

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false

```

```

# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
# Specifies the directory path to the template files in the installation
# root directory. Within the profileTemplates directory are various
# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#

```

```

# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsrv.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)

```



```

#     shortHostNameCellCellNumber
#     else
#     shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or

```

```

#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an

```

```

#      automatically generated set of recommended ports if you do not specify the
#      -startingPort parameter, the -defaultPorts parameter or the -portsFile
#      parameter. The recommended port values can be different than the default
#      port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#

```

```

# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```

```

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

```

```

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####

```



```

# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database.  The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse.  The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name.  Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password.  Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name

```

```

dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbAppMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd
#####
# Parameter: dbCeimeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd
#####
# Parameter: dbSysMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeSchemaName=sys_me_schemaName

```

```

#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####

```

```

# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: webFormInstallRoot
#
# Description:
# Specifies the full path where IBM Forms Server is installed. For example,
# the IBM Forms Server install root might be
# C:/IBM/IBMFoMServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#webFormConfig=false

```

```

#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
configureBPC=false

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

```

```

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties

```



```

# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters

```

```

# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative

```

```

# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile. Do not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to

```

```

#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck

```

```

#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter:  dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false

```

```

# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#

```



```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID

```

```

dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmbdDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmbdDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmbdDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmbdDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated

```

```

# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
#dbWinAuth=true

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample

```

```

# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with an SQL Server database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application

Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 81. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 82. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with an SQL Server database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Click *Linux_operating_system_menus_to_access_programs* > IBM > *your_product* > Profile Management Tool.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy

server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 83. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	

Table 83. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 84. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
 - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
 - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response

- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring SQL Server databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/ProcessServer/SQLServer/ at profile creation time.
2. Run the following script.
`configProcessServerDB.sh`

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Chosen not to run the database scripts to initialize the databases during the configuration.
 - Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*/dbscripts/PerformanceDW/SQLServer/ at profile creation time.
 2. Run the following script.
`configPerformanceDWDB.sh`

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the Common database during the configuration.
- Created the database that you want to create the Common database tables in.

1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/SQLServer/*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

configCommonDB.sh createDB - creates the database and tables

configCommonDB.sh - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.

1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install_root/profile/dbscripts/BusinessSpace/node_name_server_name/SQLServer/CMNDB*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:

configBusinessSpaceDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:

- a. **createDatabase.sql**
- b. **createTable_BusinessSpace.sql**
- c. **createGrant_BusinessSpace.sql**

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Important: If the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
```

```
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Modifying an existing installation:

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

Configuring a network deployment environment on Linux

For a network deployment configuration, you install the software and then create or augment a deployment manager, create or augment at least one custom profile, and configure the network deployment environment.

Granting write permission of files and directories to nonroot users for profile creation or augmentation:

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root/profiles* directory of the product.

Restrictions:

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
 - a. Create a group named **profilers**, which will contain all users who can create profiles.
 - b. Create a user named **user1**, who can create profiles.
 - c. Add users **product_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
 - Create the *install_root/logs/manageprofiles* directory:
`mkdir install_root/logs/manageprofiles`
 - Create the *install_root/properties/fsdb* directory:
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the `profileRegistry.xml` file as the installer. For this example, the file path is:
`install_root/properties/profileRegistry.xml`

Add the following information to the `profileRegistry.xml` file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **/opt/IBM/WebSphere/AppServer**.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

The root user and the nonroot user can use the same tasks to manage profiles.

Creating databases and database design files:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. You must create the databases before network deployment.

Creating databases and database design files for DB2:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

Creating DB2 databases:

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the /etc/profile file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root user to the /etc/group administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103:db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
- When the DB2 database manager is not running:
SQL1032N No start database manager command was issued. SQLSTATE=57019
- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:
SQL1092N "ROOT" does not have the authority to perform the requested command.

Creating database design files for DB2:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```


-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- ```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5)wesb.nd.topology
- (6)wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI\_BPC : [status = not complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSpace] WBI\_BSPACE : [status = not complete]
- (6)[SibME] WBI\_BPC\_ME : [status = not complete]
- (7)[SibME] WBI\_CEI\_ME : [status = not complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:



[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

- (1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]
```

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.nd is complete with 0 remaining item(s):
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with DB2 in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm\_performancedw
- (4)bpm\_processerver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processerver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :  
Database User name[default=] :  
Schema name[default=] :  
Regular pagesize[default=32k] :  
Regular TableSpace[default=WBISPACE] :  
Temporary pagesize[default=32k] :  
Temporary TableSpace[default=WBITEMPSpace] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=50000] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment



*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### **Required property is empty errors**

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### **Sample output of running a validation of the existing database design**

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### **Contents of the database design tool log file**

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating database design files and database scripts (Linux network deployment):*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files and database scripts for deployment environments with DB2 for z/OS:*

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).



**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include ".nd."

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and

press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, `WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB`. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
```

```

Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
```

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

**Note:** For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM\_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1)false
- (2>true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.nd.topology**, after you configure the master component, you must manually configure the **BPM\_PerformanceDW** and **BPM\_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :12

```
[status] bpm.advanced.nd.topology is not complete with 6 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'catalog' for VCAT is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminUserName' for SECURITY_ADMIN_USER is empty.
[4] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
[5] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[6] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'catalog' for VCAT is empty.
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment:*

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).
```



-v db\_design\_file | db\_scripts\_output\_directory  
when a db\_design\_file is given, validation will be done on the specified database design file based on the database specs.  
When a db\_scripts\_output\_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db\_design\_file [-d output\_directory] [db\_design\_file2] [-d output\_directory2] ... [db\_design\_fileN] [-d output\_directoryN]  
generate the database scripts from the specified design files in batch mode.  
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :  
Data source user name[default=] :  
Database schema name (SQLID)[default=] :  
Bufferpool of 4K size for tables[default=BP1] :  
Bufferpool for indexes[default=BP2] :  
Bufferpool of 4K size for LOB data[default=BP3] :  
Bufferpool of 8K size for tables[default=BP8K1] :  
Bufferpool of 16K size for tables[default=BP16K1] :  
Bufferpool of 32K size for tables[default=BP32K1] :  
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:



[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- 8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:  

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```
- 9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:  

```
generate database scripts? (y/n) [default=y] :
```
- 10. To generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:  

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

#### Related tasks:

“Creating the messaging engine data stores” on page 802

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the

messaging engine data stores.

### *Generating scripts for creating database objects in DB2 for z/OS:*

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
  - Database server details
  - Data source details
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and

generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: cd /opt/IBM/WebSphere/AppServer/util/dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db\_design\_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

**Tip:** In the database design file, the directories for the SQL scripts are defined by the *ddl\_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl\_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.  
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

**Note:** The `createDB2.sh` script requires execute permissions.

### *Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### **Required property is empty errors**

When the required `userName` and `password` properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### **Sample output of running a validation of the existing database design**

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### **Contents of the database design tool log file**

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

### *Setting up databases for Oracle:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a `BPMDBA` user name for the Process Server database, a `PDWDBA` user name for the Performance Data Warehouse database, and a `CMNDBA` user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

*Creating database design files for Oracle:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for deployment environments with Oracle:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.
```

```
-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).
```

```
-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
```

When a db\_scripts\_output\_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode.

The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
```



- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI\_BPC : [status = not complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = not complete]
- (6)[SibME] WBI\_BPC\_ME : [status = not complete]
- (7)[SibME] WBI\_CEI\_ME : [status = not complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server



7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict.

If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.nd.topology**, you must manually configure the **BPM\_PerformanceDW**, **BPM\_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

```
Please enter the number for the database component :3
```

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

-----  
[status] BPM\_ProcessServer is not complete with 3 remaining item(s):  
[ 1 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'databaseUser' for DB\_USER is empty.  
[ 2 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminUserName' for SECURITY\_ADMIN\_USER is empty.  
[ 3 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminPassword' for SECURITY\_ADMIN\_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

[info] The script(s) have been generated in  
WAS\_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI\_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named dbDesignGenerator.log is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with DB2 in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
```

When a `db_scripts_output_directory` is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode.  
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=CMNDB] :

Database User name[default=] :

Database schema [default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

Database Location(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)



- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.



## Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

## Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

## Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating databases and database design files for SQL Server:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

**Tip:** If you are using the federated repositories as a user registry, you can ignore warnings in the `systemout.log` file about maximum key length: ... **Warning! The maximum key length is 900 bytes ...** If you are using the stand-alone LDAP registry, ensure that the number of characters in all the user distinguished name (DN) entries in your organization do not exceed the 131 character limit. If the number of characters in any of the user DN entries exceeds 131 characters, you must change the user account registry to the federated repositories option.

*Creating SQL Server databases:*

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
  - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db\_user\_account** and **db\_user\_password** are the user name and password that are used to log in to create the database, and **database\_name** is the name of the database that you are creating. **COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

**Important:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- From the Design view, press F4 to view the Properties window.
- From the Properties window, update the schema name.
- Right-click the tab and select **Close** to close the Design view.
- Click **OK** when prompted to save. The selected table is transferred to the schema.

- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 85.

| Database management system | Minimum privilege required to use the data store tables                                                                                                                                                                                        | Additional privilege required to create the data store tables |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| Microsoft SQL Server       | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege.    |

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ\_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

*Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

**On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

**On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.

- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
  - f. Click **OK** again to close the Properties window, and then close Component Services.
  - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
  - b. Extract the archive to any folder.
  - c. Copy the sqljdbc\_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc\_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc\_xa.dll file in the x64 folder.
  - d. Run the xa\_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc\_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
  - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

*Creating database design files for SQL Server:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for deployment environments with SQL Server:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
```

help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1) bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.

[info] Completed database components can be edited to change existing or defaulted property values.

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI\_BPC : [status = not complete]
- (3) [BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4) [BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI\_BSPACE : [status = not complete]
- (6) [SibME] WBI\_BPC\_ME : [status = not complete]
- (7) [SibME] WBI\_CEI\_ME : [status = not complete]
- (8) [SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9) [SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10) [SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11) [SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI\_CommonDB is not complete with 1 remaining item(s):

[ 1 ] WBI\_CommonDB.WBI\_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.



You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:



[info] Please enter the values for the properties in the data source properties section.  
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the  
databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :  
Data source password[default=] :  
MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI\_CommonDB is complete with 0 remaining item(s):

-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components,  
since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the **BPM\_ProcessServer**, you must manually configure the **BPM\_PerformanceDW** component for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[BPM\_ProcessServer] BPM\_ProcessServer : [master] [status = complete]
- (2)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (3)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (4)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (5)[save and exit]

Please enter the number for the database component :2

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with SQL Server in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :

Data source password[default=] :

MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```



## Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

### Configuring the network deployment environment using the `configureNode` command:

To configure a typical network deployment environment, you can use the `configureNode` command instead of using the Profile Management Tool and Deployment Environment wizard or the `manageprofiles` and `wsadmin` commands. This command provides limited customization by using a properties file, and creates a ready-to-use environment. You can also use `configureNode` later to extend the deployment environment.

**Restriction:** This command is not supported if your database type is DB2 for z/OS.

**Important:** The `configureNode` command cannot be run twice on one machine without first deleting the profile that was created. To run the `configureNode` command a second time, first delete the profile.

*Creating the deployment environment using the `configureNode` command:*

You can use the `configureNode` command to create a typical network deployment environment. The goal of this command is to provide a ready-to-use environment. The only parameters that can be customized are provided in a properties file.

**Restriction:** This command is not supported if your database type is DB2 for z/OS.

You must have installed the product and created all the databases that are specified in the properties file.

**Important:** You must have already installed the database software and created the databases (but do not create tables). You must also have created all the users that you specify in the properties file. Make sure the users have the installation privileges listed in Database privileges.

For your SQL Server database server:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
- If connections to the database will be made by the current Windows user that the server is running under, the SQL Server must have **Windows authentication mode** or **SQL Server and Windows Authentication mode** enabled, as specified through Microsoft SQL Server Management Studio.

**Restriction:** Run `configureNode` commands sequentially; do not try to run the command on multiple machines at the same time.

The `configureNode` command uses default ports to create the deployment manager. You cannot customize the ports using this command. The parameters listed in the sample files are the only parameters that can be customized. Make sure that no other profiles that are configured with default ports are started on the same machine.

The `configureNode` command creates the following network deployment topology:

- A single cell environment
- The environment contains the number of cluster members that you specify in the properties file
- The environment uses the Remote Messaging, Remote Support, and Web topology pattern, which includes the following four clusters:
  - Messaging infrastructure cluster



- Supporting infrastructure cluster
- Application deployment target cluster
- Web application cluster

The command performs the following tasks:

- Creates the deployment manager node based on the values in the deployment manager properties file and starts the deployment manager.
- Creates the deployment environment definition.
- Creates a custom node based on the values in the custom node properties file.
- Federates the node and adds the node to the deployment environment.
- Generates the deployment environment.
- Creates the database tables. (You must already have created the databases. See "Before you begin" for instructions.)
- Runs the bootstrap utility to load the Process Server database with system information.
- Configures Business Space and Process Portal on the web application cluster.

The default port is used to create the deployment manager if the optional ports are not specified in the properties file. If a deployment manager is already running on the machine when you run **configureNode**, creating the deployment environment fails. If a deployment manager has already been created but is not running, you must change the port of one of the deployment managers or you might get a port conflict error when you start the other deployment manager.

To create the deployment environment for the first time, complete the following steps:

1. On the machine where you want to create the deployment environment, locate the sample properties file: *install\_root/util/ndUtils/samples/*.
2. Copy the sample files and modify the files to reflect your environment. For example, for a deployment manager for IBM BPM Advanced for Process Server, choose the *sample\_adv\_pc\_dmgr.properties* file.

For each database that you want to create, copy the database parameters section and specify the database name, user name, password, and schema name. The Process Server database, Performance Data Warehouse database, and Common database are required, and they must have different names. For IBM BPM Advanced, the Business Process Choreographer database is also required.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

**Restriction:** The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the names of the two files you created. The custom node file is optional, but if you do not include it, only the deployment manager is created. For example:

```
install_root/util/ndUtils/configureNode.sh -dmgr_response my_dmgr_response_file.properties
-response node01_response_file.properties
```

**Note:** If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to **127.0.0.1 localhost** in the hosts file. For example: **127.0.0.1 localhost example.ibm.com**.

You now have a deployment manager, a custom node, and a deployment environment that use the Remote Messaging, Remote Support, and Web topology pattern.

Messages are recorded in the file *install\_root/logs/config/configureNode.log*.

**Tip:** After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception

`com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT.` This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

**Important:** If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

**Related concepts:**

 Virtual hosts

**Related tasks:**

 Configuring virtual hosts

**Related reference:**

 Test connection service

*Extending the deployment environment using the `configureNode` command:*

After you have created a deployment environment using the **`configureNode`** command, you can run the command at a later time to extend the deployment environment.

You need the following prerequisites before running the **`configureNode`** command:

- You must have installed the product on the machine where you want to extend the deployment environment.
- The deployment manager must be running.
- You must have created the deployment environment by running the **`configureNode`** command.

**Restriction:** Run **`configureNode`** commands sequentially; do not try to run the command on multiple machines at the same time.

1. On the machine where you want to extend the deployment environment, locate the sample custom node files: `install_root/util/ndUtils/samples/`
2. Copy the sample file that is most similar to the properties file that you need. For example, to extend the deployment environment for IBM BPM Advanced for Process Server, choose the `sample_adv_ps_node.properties` file.

Modify the new custom node file to reflect your environment, including the details of the deployment manager.

For more information about the available properties, read the comments in the sample files, or see the **`configureNode`** command reference and the examples.

**Restriction:** The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **`configureNode`** command, passing it the name of the sample file. For example:

`install_root/util/ndUtils/configureNode.sh -response node01_response_file.properties`

**Note:** If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to `127.0.0.1 localhost` in the hosts file. For example: `127.0.0.1 localhost example.ibm.com`.

You have extended the deployment environment.

Messages are recorded in the file `install_root/logs/config/configureNode.log`.

**Tip:** After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception `com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT`. This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

**Important:** If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

**Related concepts:**

 [Virtual hosts](#)

**Related tasks:**

 [Configuring virtual hosts](#)

**Related reference:**

 [Test connection service](#)

**Configuring profiles and creating a network deployment environment:**

After you install the product, you must create or augment a deployment manager and one or more custom profiles to define the runtime environment. Before starting the deployment manager, you must have configured the databases that are to be used with IBM Business Process Manager.

*Creating or augmenting network deployment profiles:*

If you used the **configureNode** command to create and configure your deployment environment, no further configuration is required. Otherwise, you must create or augment a deployment manager profile and one or more custom profiles before creating the deployment environment. Using profiles, you can have more than one runtime environment on a system, without having to install multiple copies of IBM Business Process Manager.

*Creating or augmenting network deployment profiles with a DB2 database server:*

You can configure a network deployment environment for IBM Business Process Manager using a DB2 database server.

*Preparing the DB2 database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components. You can either create a new database and tables or postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

If you plan to configure the Common database on a remote server, you must have created the database before beginning to create or augment the profile. You can create a database on the local server or use an existing one on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root*/bin/ProfileManagement/pmt.sh.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root*/profiles/*profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.



These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services. You must specify a user name under which the service runs. To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step



15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 86. Required Common database configuration fields for DB2 Database

| Field                                               | Action needed                                                                                                                         |
|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| User name to authenticate with the database         | Enter the user name to authenticate with the database.                                                                                |
| Password for database authentication                | Enter a password to authenticate with the database.                                                                                   |
| Confirm password                                    | Confirm the password.                                                                                                                 |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdriivers/DB2</code> . |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                 |
| Server port                                         | Accept the default value of 50000 or enter the correct server port number.                                                            |
| Schema name                                         | Enter a schema name for the Common database.                                                                                          |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with a DB2 database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
- Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
- Run the command `install_root/bin/ProfileManagement/pmt.sh`.

2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.

5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.

7. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The

password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step
15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 87. Required Common database configuration fields for DB2 Database

| Field                                               | Action needed                                                                                                          |
|-----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| User name to authenticate with the database         | Enter the user name to authenticate with the database.                                                                 |
| Password for database authentication                | Enter a password to authenticate with the database.                                                                    |
| Confirm password                                    | Confirm the password.                                                                                                  |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/jdbcdrivers/DB2. |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                  |
| Server port                                         | Accept the default value of 50000 or enter the correct server port number.                                             |
| Schema name                                         | Enter a schema name for the Common database.                                                                           |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.



19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create a deployment manager profile.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response

- PC\_Adv\_DMgr\_Oracle.response
- PC\_Adv\_DMgr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMgr\_DB2.response
- PS\_Std\_DMgr\_DB2zOS.response
- PS\_Std\_DMgr\_Oracle.response
- PS\_Std\_DMgr\_SQLServer.response
- PS\_Adv\_DMgr\_DB2.response
- PS\_Adv\_DMgr\_DB2zOS.response
- PS\_Adv\_DMgr\_Oracle.response
- PS\_Adv\_DMgr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```



Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2.response
```

```

#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####

```

profileName=Dmgr01

```

Parameter: cellName

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: nodeName

Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

cellName=cell_name
nodeName=node_name

Parameter: enableAdminSecurity

Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.

Valid Values:
true
Default Values:
true

enableAdminSecurity=true
adminUserName=adminUser_ID
```

adminPassword=adminPassword

```

Parameter: signingCertDN

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

Valid Values:
Varies
Default Values:
None

Parameter: signingCertValidityPeriod

Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

Valid Values:
Varies
Default Values:
None

Parameter: keyStorePassword

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.

Valid Values:
Varies
Default Values:
None

#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

Parameter: defaultPorts

Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

Valid Values:
Varies
Default Values:
None

Parameter: startingPort

Description:
```

```

Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:

```

```

Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:

```

```

false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of

```



```

database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the

```

```

database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be

```

```

accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

```

```

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties

```

```

file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword

```

```

along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#

```



```

Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winServiceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceCheck
#
Description:

```

```

The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####

```

```

Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.

```

```

#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else

```

```

shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the

```



```

profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual

```

```

automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#

```

```

Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#

```

```

Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:

```

```

Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file

```

```

is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName

```

```

#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN

```



```

#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and

```

```

resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#

```

```

Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:

```

```

false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with a DB2 database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. If you selected **Typical** profile creation, skip to the Administrative Security step.
8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
9. **Advanced:** On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. **Required:** On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.



Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 88. Required Common database configuration fields for DB2 Database

| Field                                               | Action needed                                                                                                                        |
|-----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| User name to authenticate with the database         | Enter the user name to authenticate with the database.                                                                               |
| Password for database authentication                | Enter a password to authenticate with the database.                                                                                  |
| Confirm password                                    | Confirm the password.                                                                                                                |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> . |
| Database server host name (for example IP address)  | Accept the default value of <code>localhost</code> or enter the correct database server host name.                                   |
| Server port                                         | Accept the default value of <code>50000</code> or enter the correct server port number.                                              |
| Schema name                                         | Enter a schema name for the Common database.                                                                                         |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with a DB2 database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the `java user.language` setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.  
If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root

signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.



To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 89. Required Common database configuration fields for DB2 Database

| Field                                               | Action needed                                                                                                                         |
|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| User name to authenticate with the database         | Enter the user name to authenticate with the database.                                                                                |
| Password for database authentication                | Enter a password to authenticate with the database.                                                                                   |
| Confirm password                                    | Confirm the password.                                                                                                                 |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdriivers/DB2</code> . |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                 |
| Server port                                         | Accept the default value of 50000 or enter the correct server port number.                                                            |
| Schema name                                         | Enter a schema name for the Common database.                                                                                          |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.



4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response

- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the *bin* directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default.

The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response



- PC\_Std\_Standalone\_SQLServer.response
- PC\_Adv\_Standalone\_DB2.response
- PC\_Adv\_Standalone\_DB2zOS.response
- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```



The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
```

```

directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```

```

Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the

```

```

root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the

```

```

user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled

```

```

Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.

```

```

Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values

```



```

are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a

```

```

Managed PC Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be

```

```

currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#

```

```

Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the

```

```

deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.

```



```

Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#

```

```

To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr.templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.

```

```

#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#

```

```

Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER

```

```
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
```

```

Parameter: federateLaterBPM

Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.

Valid Values:
false
true
Default Values:
true

Parameter: dmgrHost

Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.

Valid Values:
Varies
Default Values:
None

Parameter: dmgrPort

Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
```



```

the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
```

```

#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:

```

```

if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile

```

```

creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: federateLaterBPM

```

```

#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:

```



```

Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones.

If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### **Related information:**

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.

- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response



- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:



- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
  - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:
 

```
profile_root/bin/startManager.sh
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):
 

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```

An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### Related information:



Add managed node settings (WebSphere Application Server)

*Creating or augmenting network deployment profiles with a DB2 for z/OS database server:*

You can configure a network deployment environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
  - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other

WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.

- d. Click **Next**. The design file location is passed to the command line to complete the database configuration.

15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.

17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.

18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Run the scripts to create and configure the DB2 for z/OS database.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool on Linux:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
  - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.



- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.

- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
 

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.



15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:
  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response

- PC\_Adv\_DMgr\_Oracle.response
- PC\_Adv\_DMgr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMgr\_DB2.response
- PS\_Std\_DMgr\_DB2zOS.response
- PS\_Std\_DMgr\_Oracle.response
- PS\_Std\_DMgr\_SQLServer.response
- PS\_Adv\_DMgr\_DB2.response
- PS\_Adv\_DMgr\_DB2zOS.response
- PS\_Adv\_DMgr\_Oracle.response
- PS\_Adv\_DMgr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
```

```

was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true

```

```
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
```

```

Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#

```



```

Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true

```

```

Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of

```

```

database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the

```

```

-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage

```

```

Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

```

```

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each

```

```

profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you

```



```

create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.

```

```

During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual

```

```

automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true

```

```
Default Values:
false
#####
#isDeveloperServer=false
```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
```

```

When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.

```

```

Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies

```



```

Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign

```

```

#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:

```

```

false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:

```

```

The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
```



```

Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```

```

The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```

```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a

```

```

stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
  - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  8. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:



- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

*profile\_root/properties/portdef.prop*

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

14. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

15. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.

19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
  - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
8. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page
 When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

14. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.



You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

15. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - *dmgr.procctr*: for a Process Center deployment manager profile for IBM BPM Standard.
  - *dmgr.procctr.adv*: for a Process Center deployment manager profile for IBM BPM Advanced.
  - *dmgr.procsvr*: for a Process Server deployment manager profile for IBM BPM Standard.
  - *dmgr.procsvr.adv*: for a Process Server deployment manager profile for IBM BPM Advanced.
  - *dmgr.esbserver*: for a WebSphere Enterprise Service Bus deployment manager profile.

3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2z0S.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`
- `PS_Adv_DMGr_DB2.response`
- `PS_Adv_DMGr_DB2z0S.response`



- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

#### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

#### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.

- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response

- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:



**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
```



```

Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:

```

```

if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root

```

```

signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType

```

```

#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom

```

```

profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:

```

```

Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT
#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses DB2zOS.

```



```

#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:

```

```

true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log

```

```

install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs

```

```

the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage

```

```

Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:

```



```

#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2z0S.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####

```

```
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
```

```

Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winServiceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceCheck
#
Description:

```

```

The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.

```

```

#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

```

```

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied

```



```

and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a

```

```

unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name

```

nodeName=node\_name

```

Parameter: enableAdminSecurity

Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.

Valid Values:
true
Default Values:
true

enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

Parameter: signingCertDN

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

Valid Values:
Varies
Default Values:
None

Parameter: signingCertValidityPeriod

Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

Valid Values:
Varies
Default Values:
None

Parameter: keyStorePassword

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.

Valid Values:
Varies
Default Values:
None

#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

Parameter: enableService
```

```

#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true

```

```

Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:

```

```

Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:



```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for

commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

#### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

#### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### Related information:



Updating ports in existing profiles (WebSphere Application Server)

*Augmenting custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:
  - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
  - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
  - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
  - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
  - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response

- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:  

```
profile_root/bin/startManager.sh
```
- The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
- The deployment manager is at the same release level or higher than the custom profile you created or augmented.



- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

*Creating and configuring DB2 for z/OS databases after network deployment profile creation:*

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*DB2 for z/OS authorization prerequisites:*

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

#### User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as *#DB\_USER#*. These permissions are available in the *createDatabase.sql* database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the *#DB\_USER#* user to create sequences and stored procedures with a schema qualifier of *#SCHEMA#*:



```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

### Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database #DB\_NAME# can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;
```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named SICELL, and connect user WSADMIN to it, as follows:

```
INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
 WSADMIN CONNECT 000000 NONE
 CONNECT ATTRIBUTES=NONE
 REVOKE DATE=NONE RESUME DATE=NONE
```

## Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

## Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

## Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

### Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832  
Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating databases in the DB2 for z/OS subsystem:*

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

### Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

### Considerations for choosing the createDB2.sh script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

### Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two .sql files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

### Related tasks:

“Creating database design files and database scripts (Linux network deployment)” on page 565  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Configuring the DB2 command line processor:*

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
  - Modify the CLASSPATH environment variable to include the `clp.jar` file.
  - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
  - Define the **db2** command as an alias for the command that starts the command line processor.
  - Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:

- A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
- A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

**Tip:** When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

#### Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script”

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 798

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (*z\_schema.sql* and *z\_schemaProc.sql*) that you can run later by using the DB2 command line processor.

*Creating DB2 for z/OS database objects using the createDB2.sh script:*

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTPE2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database



configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCcreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**-RunSQL**

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

**Tip:** The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 795

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using the DB2 command line processor:*

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database



configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCcreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**Tip:** By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two .sql files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

The z\_schema.sql and z\_schemaProc.sql files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The z\_schemaProc.sql file contains SQL statements for stored procedures, and is generated in addition to z\_schema.sql because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the z\_schema.sql file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the z\_schemaProc.sql file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

#### **Related tasks:**

“Configuring the DB2 command line processor” on page 795

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:*

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

**Tip:** The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

**SPUFI** A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

### DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:  

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

**Important:** After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

**Tip:** If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

```
@STOGRP@ = DB2 Storage group name
@DB_NAME@ = DB2 Database name
@SCHEMA@ = DB2 SQLID (Schema Qualifier)
@BP4K@ = Buffer Pool of 4k Size
@BPINDEX@ = Buffer Pool Index
@VOLUMES@ = DB2 comma separated list of volume-ids
@VCAT@ = DB2 integrated catalog facility name
@DB_USER@ = The user ID which will be used to create the database objects
```

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

**Tip:** If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (z\_schema.sql and z\_schemaProc.sql), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

### Related tasks:

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 798

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z\_schema.sql and z\_schemaProc.sql) that you can run later by using the DB2

command line processor.

*Creating the messaging engine data stores:*

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Before you create and run the database scripts:

- Create a cluster.
- Decide which service integration bus components you want to create in your network deployment configuration.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
  - When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
    - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
    - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
    - a. Run the database design tool (**DbDesignGenerator**) from the *install\_root/util/dbUtils* directory.
    - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
    - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

**Tip:** As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

**Related tasks:**

“Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment” on page 572

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for deployment environments with DB2 for z/OS” on page 565

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

*Granting table privileges to the JCA authentication alias user ID:*

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

**Note:** Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

*Setting the correct schema name for the messaging engines:*

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
  - a. Select **Messaging engines**, and then click the name that is displayed.
  - b. Click **Message store**.
  - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
  - d. Click **Apply**.
  - e. Save your configuration changes.
4. Verify that the messaging engines can be started:

- a. Log out of the administrative console.
- b. Stop, and then restart the server.
- c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

*Creating or augmenting network deployment profiles with an Oracle database server:*

You can configure a network deployment environment for IBM Business Process Manager using an Oracle database server.

*Preparing the Oracle database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

To configure the Common database while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default



language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.



- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 90. Required database configuration fields for Oracle - Part 2

| Field                                               | Action needed                                                                                                                                                           |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database user name                           | Enter the user name to authenticate with the database.                                                                                                                  |
| Common database password                            | Enter a password to authenticate with the database.                                                                                                                     |
| Confirm password                                    | Confirm the password.                                                                                                                                                   |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                                                   |
| Server port                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                               |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.

- Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The



password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 91. Required database configuration fields for Oracle - Part 2

| Field                                               | Action needed                                                                                                                                                           |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database user name                           | Enter the user name to authenticate with the database.                                                                                                                  |
| Common database password                            | Enter a password to authenticate with the database.                                                                                                                     |
| Confirm password                                    | Confirm the password.                                                                                                                                                   |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                                                   |
| Server port                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                               |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.



18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:
  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response

- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
```

```

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:

```

```

Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:

```

```

Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.

```

```

Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true, the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies

```



```

Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:

```

```

manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true

```

```

Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:

```

```

The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_passwd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false
#####

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
```

```

Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```



```

The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```

```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use

```

```

type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
```

```

Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```



```

The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```

```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous

```

```

versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.

```



```

Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.

```

```

#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:

```

```

true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None

```

```

#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#

```

```

Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production

```



```

Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with an Oracle database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).

- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
 

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the *bin* directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 92. Required database configuration fields for Oracle - Part 2

| Field                                               | Action needed                                                                                                                                                           |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database user name                           | Enter the user name to authenticate with the database.                                                                                                                  |
| Common database password                            | Enter a password to authenticate with the database.                                                                                                                     |
| Confirm password                                    | Confirm the password.                                                                                                                                                   |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                                                   |
| Server port                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                               |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.

- Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with an Oracle database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.



**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
  11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
    - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
    - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:



- In the **Path** field, enter the directory path to the existing certificate.
- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

*profile\_root/properties/portdef.prop*

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.

18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 93. Required database configuration fields for Oracle - Part 2

| Field                     | Action needed                                          |
|---------------------------|--------------------------------------------------------|
| Common database user name | Enter the user name to authenticate with the database. |

Table 93. Required database configuration fields for Oracle - Part 2 (continued)

| Field                                               | Action needed                                                                                                                                                           |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database password                            | Enter a password to authenticate with the database.                                                                                                                     |
| Confirm password                                    | Confirm the password.                                                                                                                                                   |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                                                   |
| Server port                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                               |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.

3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2z0S.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`
- `PS_Adv_DMGr_DB2.response`
- `PS_Adv_DMGr_DB2z0S.response`

- PS\_Adv\_DMgr\_Oracle.response
- PS\_Adv\_DMgr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.



8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.



- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response

- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
```

```

Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:

```

```

if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root

```

```

signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.

```



```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType

```

```

#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is

```

```

8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true

```

```
Default Values:
false
#####
#isDeveloperServer=false
```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
```

```

path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile

```

```

that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:

```



```

None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:

```

```

localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID

```

```

#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the

```

```

-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer

```

```

#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

```

```

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is

```

```

the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies

```



```

Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic

```

```

disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

```

```

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
```

```

#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:

```



```

if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile

```

```

creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbDriverType

```

```

#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when

```

```

creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

8. If you selected **Typical** profile creation, skip to the Federation step.

9. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.



- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### **Related information:**

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the `manageprofiles` command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available for custom profiles:

- *managed.procctr*: for a Process Center custom profile for IBM BPM Standard.
- *managed.procctr.adv*: for a Process Center custom profile for IBM BPM Advanced.
- *managed.procsvr*: for a Process Server custom profile for IBM BPM Standard.
- *managed.procsvr.adv*: for a Process Server custom profile for IBM BPM Advanced.
- *managed.esbserver*: for a WebSphere Enterprise Service Bus custom profile.

3. Use the **augment** parameter to make changes to an existing profile with an augmentation template. The **augment** parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- *PC\_Std\_DMGr\_DB2.response*
- *PC\_Std\_DMGr\_DB2zOS.response*
- *PC\_Std\_DMGr\_Oracle.response*
- *PC\_Std\_DMGr\_SQLServer.response*
- *PC\_Adv\_DMGr\_DB2.response*
- *PC\_Adv\_DMGr\_DB2zOS.response*
- *PC\_Adv\_DMGr\_Oracle.response*
- *PC\_Adv\_DMGr\_SQLServer.response*
- *PC\_Std\_Managed\_DB2.response*
- *PC\_Std\_Managed\_DB2zOS.response*
- *PC\_Std\_Managed\_Oracle.response*
- *PC\_Std\_Managed\_SQLServer.response*
- *PC\_Adv\_Managed\_DB2.response*

- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2z0S.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2z0S.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
 

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
  - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:  

```
profile_root/bin/startManager.sh
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

**Related information:**

 Add managed node settings (WebSphere Application Server)

*Creating or augmenting network deployment profiles with an SQL Server database server:*

You can configure a network deployment environment for IBM Business Process Manager using a Microsoft SQL Server database server.

*Preparing the SQL Server database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

**Restriction:** The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

*Table 94.*

| Database management system | Minimum privilege required to use the data store tables                                                                                                                                                                                        | Additional privilege required to create the data store tables |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| Microsoft SQL Server       | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege.    |

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ\_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

*Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:



**On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

**On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
  - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
  - c. Right-click **Local DTC** and then select **Properties**.
  - d. Click the **Security** tab on the **Local DTC Properties** window.
  - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
  - f. Click **OK** again to close the Properties window, and then close Component Services.
  - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
  - b. Extract the archive to any folder.
  - c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the x64 folder.
  - d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
  - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```



For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root*/bin/ProfileManagement/pmt.sh.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root*/profiles/*profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 95. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                                                                               | Action needed                                                                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. |                                                                                                                                                                                      |
| <b>Common database user name</b>                                                                                                                                                                                                                                                                    | Enter the user name to authenticate with the database.                                                                                                                               |
| <b>Common database password</b>                                                                                                                                                                                                                                                                     | Enter a password to authenticate with the database.                                                                                                                                  |
| <b>Confirm password</b>                                                                                                                                                                                                                                                                             | Confirm the password.                                                                                                                                                                |
| <b>Location (directory) of JDBC driver classpath files</b>                                                                                                                                                                                                                                          | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |
| <b>Database server host name (for example IP address)</b>                                                                                                                                                                                                                                           | Accept the default value of localhost or enter the correct database server host name.                                                                                                |
| <b>Server port</b>                                                                                                                                                                                                                                                                                  | Accept the default value of 1433 or enter the correct server port number.                                                                                                            |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.



- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
- Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.



**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 96. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                      | Action needed                                                                                                                                                                                                                                                                                       |
|------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                            | Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. |
| <b>Common database user name</b>                           | Enter the user name to authenticate with the database.                                                                                                                                                                                                                                              |
| <b>Common database password</b>                            | Enter a password to authenticate with the database.                                                                                                                                                                                                                                                 |
| <b>Confirm password</b>                                    | Confirm the password.                                                                                                                                                                                                                                                                               |
| <b>Location (directory) of JDBC driver classpath files</b> | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer                                                                                                                             |
| <b>Database server host name (for example IP address)</b>  | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                               |
| <b>Server port</b>                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                                                                                                                                                           |

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response

- PC\_Std\_Standalone\_SQLServer.response
- PC\_Adv\_Standalone\_DB2.response
- PC\_Adv\_Standalone\_DB2zOS.response
- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDbName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a deployment manager profile using the **manageprofiles** command that includes the following components: CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space.

```
install_root/bin/manageprofiles -create -templatePath install_root\profileTemplates\BPM\
\dmgr.procsvr.adv -dbHostName hostName -dbServerPort 1433 -dbDelayConfig true
-configureBSpace true -dbType MSSQLSERVER_Microsoft -dbUserId userID -dbJDBCClasspath
path to JDBC 2.0 driver -dbName CMNDB
-dbPassword password
```

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.

END COPYRIGHT

These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses SQLServer.

Depending on your environment, you may need to change the default values.

To create a profile with this response file specify:

was.install.root/bin/manageprofiles -response PC_Adv_DMgr_SQLServer.response

If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
```

```

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:

```

```

Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:

```

```

Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.

```



```

Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true, the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:

```

```

manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentType

```

```

#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own

```

```

instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces

```



```

or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNameewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

```

```

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#

```

```

Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true, the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux

```

```

service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or

```

```

test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.

```

```

#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the

```



```

install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#

```

```

Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an

```

```

automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log

```

```

install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbidbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#

```

```

Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####

```



```

Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true

```

```
Default Values:
false
#####
#isDeveloperServer=false
```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
```

```

When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.

```

```

Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbWinAuth

```



```

#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with an SQL Server database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
  11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
    - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
    - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
      - In the **Path** field, enter the directory path to the existing certificate.

- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service.

If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server as a Linux service and issue the **startServer** command, the **wasservice** command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 97. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                                                                               | Action needed                                          |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. |                                                        |
| <b>Common database user name</b>                                                                                                                                                                                                                                                                    | Enter the user name to authenticate with the database. |
| <b>Common database password</b>                                                                                                                                                                                                                                                                     | Enter a password to authenticate with the database.    |
| <b>Confirm password</b>                                                                                                                                                                                                                                                                             | Confirm the password.                                  |

Table 97. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

| Field                                               | Action needed                                                                                                                                                           |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address)  | Accept the default value of localhost or enter the correct database server host name.                                                                                   |
| Server port                                         | Accept the default value of 1433 or enter the correct server port number.                                                                                               |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with an SQL Server database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).



- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the *bin* directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.



- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: If you do not have root privileges, skip to the next step. If you have root privileges, on the Service Definition page, indicate whether to use a Linux service to run IBM Business Process Manager. By default, IBM Business Process Manager is not selected to run as a Linux service. If the profile is configured as a Linux service, IBM Business Process Manager attempts to start Linux services for processes that are started by the `startServer` or `startManager` commands. For example, if you configure a server as a Linux service and issue the `startServer` command, the `wasservice` command starts the defined services.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have the required privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 98. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                                                                               | Action needed                                                                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. |                                                                                                                                                                                      |
| <b>Common database user name</b>                                                                                                                                                                                                                                                                    | Enter the user name to authenticate with the database.                                                                                                                               |
| <b>Common database password</b>                                                                                                                                                                                                                                                                     | Enter a password to authenticate with the database.                                                                                                                                  |
| <b>Confirm password</b>                                                                                                                                                                                                                                                                             | Confirm the password.                                                                                                                                                                |
| <b>Location (directory) of JDBC driver classpath files</b>                                                                                                                                                                                                                                          | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |
| <b>Database server host name (for example IP address)</b>                                                                                                                                                                                                                                           | Accept the default value of localhost or enter the correct database server host name.                                                                                                |
| <b>Server port</b>                                                                                                                                                                                                                                                                                  | Accept the default value of 1433 or enter the correct server port number.                                                                                                            |

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:
  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response

- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root*/bin/ProfileManagement/pmt.sh.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.



6. If you selected **Typical** profile creation, skip to the Federation step.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.



- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

*profile\_root/properties/portdef.prop*

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
- managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
- managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.

2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response

- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2z0S.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2z0S.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
```

```

5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a

```

```

trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```

```

Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```



```

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom

```

```

node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a

```

```

preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#

```

```

was.install.root/bin/manageprofiles -response PC_Std_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true

```



```
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
```

```

false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows

```

```

service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false

```

```

true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#

```

```

Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal

```

```

use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On

```

```

Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####

```



```

Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or

```

```

false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword

```

```

#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath

```

```

#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the

```

```

profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:

```

```

Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .

```

```

#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#

```



```

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#

```

```

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName

```

```

#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false

```

```

true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#

```

```

Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify

```

```

Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click *Linux\_operating\_system\_menus\_to\_access\_programs* > IBM > *your\_product* > Profile Management Tool.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.



- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.

- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the `manageprofiles` command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:
  - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
  - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
  - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
  - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
  - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response

- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:  

```
profile_root/bin/startManager.sh
```
- The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
- The deployment manager is at the same release level or higher than the custom profile you created or augmented.

- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```

An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### Related information:



Add managed node settings (WebSphere Application Server)

#### Loading the database with system information in a network deployment environment:

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly

- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. For a cluster, you need to specify the cluster name. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.
- If a single WebSphere cell contains multiple application target clusters, you must run this command on each of the clusters.

Run the bootstrap utility from the command line. The bootstrap utility is found in the deployment manager profile directory. For example:

```
Linux UNIX install_root/profiles/Dmgr01/bin
```

```
Windows install_root\profiles\Dmgr01\bin
```

Run the bootstrap utility using one of the following commands:

- **bootstrapProcessServerData.sh -clusterName** *cluster\_name*
- **bootstrapProcessServerData.sh -nodeName** *node\_name* **-serverName** *server\_name*



where:

- **-clusterName** is the name of the application target cluster. You must specify this parameter when you want the bootstrap data to run on a cluster.
- **-nodeName** is the name of the node. You must specify this parameter and the **-serverName** parameter when you want the bootstrap data to run on the server that is part of the network deployment environment and not part of the cluster.
- **-serverName** is the name of the server. You must specify either this parameter and the **-nodeName** parameter when you want the bootstrap data to run on the server that is part of the network deployment and not part of the cluster.

The parameters are case-sensitive.

**Important: For SQL Server only:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

You have loaded the database with system information before successfully starting the IBM Business Process Manager server. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.clusterName.timestamp.log` or `bootstrapProcessServerData.nodeName.serverName.timestamp.log` depending on the target you specified. The console displays a subset of the logged information.

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData
```

Bootstrap data onto a cluster that hosts the Process Server or Process Center:

```
bootstrapProcessServerData -clusterName myAppCluster
```



## Modifying an existing installation:

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

## Installing and configuring IBM Business Process Manager on AIX

Install IBM Business Process Manager on AIX, and configure a stand-alone or network deployment environment.

### Configuring a stand-alone environment on AIX

You can use a typical installation to install the software and configure a stand-alone Process Center or Process Server profile. Or you can choose custom installation and then create profiles, giving you more control over how to configure your software.

### Typical installation and configuration:

The Typical installation option is the simplest and quickest method for installing and configuring IBM Business Process Manager.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

### Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

*Installing and configuring IBM Business Process Manager with a DB2 database server:*

You can install IBM Business Process Manager using a DB2 database server. You must make sure the required databases are available before you install.

*Creating DB2 databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

*Installing and configuring Process Center with a DB2 database server:*

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.  

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:

- **Hostname:** This field shows the name of your machine.

**Important:** If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 99. Required database configuration fields for DB2

| Field                                           | Action needed                                                                                                                              |
|-------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Hostname</b>                                 | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                         |
| <b>Port</b>                                     | Accept the default value of 50000 or enter the correct server port number.                                                                 |
| <b>Username</b>                                 | Enter the user name to authenticate with the database.<br><b>Restriction:</b> User names must not contain National Language Strings (NLS). |
| <b>Password</b>                                 | Enter a password to authenticate with the database.                                                                                        |
| <b>Common database name</b>                     | Enter the name for the Common database.                                                                                                    |
| <b>Process Center database name</b>             | Enter the name for the Process Center database.                                                                                            |
| <b>Performance Data Warehouse database name</b> | Enter the name for the Performance Data Warehouse database.                                                                                |

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

**Note:** The Process Designer is supported on the Windows platform only.

**Related information:**

 Installation Manager updates

*Installing and configuring Process Server with a DB2 database server:*

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.

5. Select **Install Process Server**.
6. Specify Process Server information:
  - **Hostname:** This field shows the name of your machine.
  - **Location:** Click **Browse** to change the installation location.
  - Environment **Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
  - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

**Restriction:** Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

*Table 100. Required database configuration fields for DB2*

| Field                | Action needed                                                                                                                              |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| Hostname             | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                         |
| Port                 | Accept the default value of 50000 or enter the correct server port number.                                                                 |
| Username             | Enter the user name to authenticate with the database.<br><b>Restriction:</b> User names must not contain National Language Strings (NLS). |
| Password             | Enter a password to authenticate with the database.                                                                                        |
| Common database name | Enter the name for the Common database.                                                                                                    |

Table 100. Required database configuration fields for DB2 (continued)

| Field                                    | Action needed                                               |
|------------------------------------------|-------------------------------------------------------------|
| Process Server database name             | Enter the name for the Process Server database.             |
| Performance Data Warehouse database name | Enter the name for the Performance Data Warehouse database. |

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

**Related information:**

 Installation Manager updates

*Installing and configuring IBM Business Process Manager with an Oracle database server:*

You can install IBM Business Process Manager using an Oracle database server. You must make sure the required databases are available before you install.

*Creating Oracle databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.



*Installing and configuring Process Center with an Oracle database server:*

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.  

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
  - **Hostname:** This field shows the name of your machine.



**Important:** If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
  8. Click **Next**.
  9. Select **Yes** to use an existing database.
  10. Specify the required database information.

Table 101. Required database configuration fields for Oracle

| Field                                                                           | Action needed                                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Hostname</b>                                                                 | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                                                                                                                                                                                                                                                       |
| <b>Port</b>                                                                     | Accept the default value of 1521 or enter the correct server port number.                                                                                                                                                                                                                                                                                                |
| <b>Database name</b>                                                            | Enter the correct Process Center database System Identifier.                                                                                                                                                                                                                                                                                                             |
| <b>Common database</b>                                                          | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common database.</li> </ul>                                                                                                    |
| <b>Process Center database</b>                                                  | For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Process Center database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Process Center database.</li> </ul>                                                                            |
| <b>Performance Data Warehouse database</b>                                      | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> </ul>                                        |
| <b>Common Event Infrastructure bus messaging engine database</b>                | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common Event Infrastructure bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.</li> </ul> |
| <b>Service Component Architecture system bus messaging engine database</b>      | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA system bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA system bus messaging engine.</li> </ul>                                                    |
| <b>Service Component Architecture application bus messaging engine database</b> | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA application bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA application bus messaging engine.</li> </ul>                                     |

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

**Note:** The Process Designer is supported on the Windows platform only.

**Related information:**

 Installation Manager updates

*Installing and configuring Process Server with an Oracle database server:*

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.  

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
  - **Hostname:** This field shows the name of your machine.
  - **Location:** Click **Browse** to change the installation location.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
  - **Environment Name:** The environment name is used to connect from a Process Center to this Process Server.

**Restriction:** Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 102. Required database configuration fields for Oracle

| Field                                                                      | Action needed                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Hostname</b>                                                            | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                                                                                                                                                                                                                                                       |
| <b>Port</b>                                                                | Accept the default value of 1521 or enter the correct server port number.                                                                                                                                                                                                                                                                                                |
| <b>Database name</b>                                                       | Enter the correct Process Center database System Identifier.                                                                                                                                                                                                                                                                                                             |
| <b>Common database</b>                                                     | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common database.</li> </ul>                                                                                                    |
| <b>Process Server database</b>                                             | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Process Server database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Process Server database.</li> </ul>                                                                            |
| <b>Performance Data Warehouse database</b>                                 | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> </ul>                                        |
| <b>Common Event Infrastructure bus messaging engine database</b>           | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common Event Infrastructure bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.</li> </ul> |
| <b>Service Component Architecture system bus messaging engine database</b> | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA system bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA system bus messaging engine.</li> </ul>                                                    |

Table 102. Required database configuration fields for Oracle (continued)

| Field                                                                    | Action needed                                                                                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Service Component Architecture application bus messaging engine database | <p>For the SCA application bus messaging engine, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA application bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA application bus messaging engine.</li> </ul> |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

**Related information:**

 [Installation Manager updates](#)

*Installing and configuring IBM Business Process Manager with an SQL Server database server:*

You can install IBM Business Process Manager using a Microsoft SQL Server database server. You must make sure the required databases are available before you install.

*Creating SQL Server databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
  - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db\_user\_account** and **db\_user\_password** are the user name and password that are used to log in to create the database, and **database\_name** is the name of the database that you are creating. **COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

**Important:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default



schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

#### *Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

#### **On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

#### **On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
  - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
  - c. Right-click **Local DTC** and then select **Properties**.
  - d. Click the **Security** tab on the **Local DTC Properties** window.
  - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
  - f. Click **OK** again to close the Properties window, and then close Component Services.
  - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
    - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
    - b. Extract the archive to any folder.



- c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the `x86` folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the `x64` folder.
- d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

*Installing and configuring Process Center with an SQL Server database server:*

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.  
`extract_directory/launchpad.sh`
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
  - **Hostname:** This field shows the name of your machine.  
  

**Important:** If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
  - **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 103. Required database configuration fields for SQL Server

| Field                                           | Action needed                                                                                                                                                                                                                                                                                                                     |
|-------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Hostname</b>                                 | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                                                                                                                                                                                                                |
| <b>Port</b>                                     | Accept the default value of 1433 or enter the correct server port number.                                                                                                                                                                                                                                                         |
| <b>Common database name</b>                     | Enter the name for the Common database.                                                                                                                                                                                                                                                                                           |
| <b>Process Center database name</b>             | Enter the Process Center database name.                                                                                                                                                                                                                                                                                           |
| <b>Performance Data Warehouse database name</b> | Enter the Performance Data Warehouse database name.                                                                                                                                                                                                                                                                               |
| <b>Common database</b>                          | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common database.</li> </ul>                                                             |
| <b>Process Center database</b>                  | For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Process Center database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Process Center database.</li> </ul>                                     |
| <b>Performance Data Warehouse database</b>      | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> </ul> |

Table 103. Required database configuration fields for SQL Server (continued)

| Field                                                                    | Action needed                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common Event Infrastructure bus messaging engine database                | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common Event Infrastructure bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.</li> </ul> |
| Service Component Architecture system bus messaging engine database      | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA system bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA system bus messaging engine.</li> </ul>                                                    |
| Service Component Architecture application bus messaging engine database | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA application bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA application bus messaging engine.</li> </ul>                                     |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

**Note:** The Process Designer is supported on the Windows platform only.

**Related information:**

 Installation Manager updates

*Installing and configuring Process Server with an SQL Server database server:*

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.  

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
  - **Hostname:** This field shows the name of your machine.
  - **Location:** Click **Browse** to change the installation location.
  - **Environment Type:** Select how the Process Server is used:

- Select **Production** if the server is to be used in a production capacity.
- Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
- Select **Test** if the server is to be used as a testing environment, for example, for load testing.
- Environment **Name**: The environment name is used to connect from a Process Center to this Process Server.

**Restriction:** Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

*Table 104. Required database configuration fields for SQL Server*

| Field                                           | Action needed                                                                                                                                                                                                                                                         |
|-------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Hostname</b>                                 | Accept the default value of localhost or enter the correct database server host name. For example, the IP address.                                                                                                                                                    |
| <b>Port</b>                                     | Accept the default value of 1433 or enter the correct server port number.                                                                                                                                                                                             |
| <b>Common database name</b>                     | Enter the name for the Common database.                                                                                                                                                                                                                               |
| <b>Process Server database name</b>             | Enter the Process Server database name.                                                                                                                                                                                                                               |
| <b>Performance Data Warehouse database name</b> | Enter the Performance Data Warehouse database name.                                                                                                                                                                                                                   |
| <b>Common database</b>                          | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common database.</li> </ul> |

Table 104. Required database configuration fields for SQL Server (continued)

| Field                                                                    | Action needed                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Process Server database                                                  | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Process Server database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Process Server database.</li> </ul>                                                                            |
| Performance Data Warehouse database                                      | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> </ul>                                        |
| Common Event Infrastructure bus messaging engine database                | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the Common Event Infrastructure bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.</li> </ul> |
| Service Component Architecture system bus messaging engine database      | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA system bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA system bus messaging engine.</li> </ul>                                                    |
| Service Component Architecture application bus messaging engine database | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name:</b> Enter the SCA application bus messaging engine user name.</li> <li>• <b>Password:</b> Enter a password to authenticate with the SCA application bus messaging engine.</li> </ul>                                     |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional



required fixes. See *Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0* for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

**Related information:**

 [Installation Manager updates](#)

**Custom installation and configuration:**

Use the Custom installation option to install IBM Business Process Manager if you need any installation or configuration options that are not provided by the Typical installation option, if you want to install silently, or if you want to install on an existing installation of WebSphere Application Server.

**Related reference:**

 [IBM Business Process Manager Advanced system requirements](#)

*Installing IBM Business Process Manager Advanced:*

With custom installation, you can select interactive or silent installation. You must select interactive installation if you want to install on an existing WebSphere Application Server.

**Related reference:**

 [IBM Business Process Manager Advanced system requirements](#)

*Installing IBM Business Process Manager Advanced interactively:*

You can install the IBM Business Process Manager product package interactively. You must choose interactive installation if you want to install on an existing WebSphere Application Server.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

**Related reference:**

 [IBM Business Process Manager Advanced system requirements](#)

*Installing interactively with a new installation of WebSphere Application Server:*

Using custom installation, you can install IBM Business Process Manager with a new installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
  - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on



your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```

- If you are installing from images downloaded from Passport Advantage, perform the following steps:
  - a. Go to the directory into which you extracted the images.
  - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
5. Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
6. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package.
7. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

8. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.

9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
  - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
  - b. Optional: Click a feature to view its brief description under **Details**.
  - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

**Restriction:** Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
12. When the installation process is complete, a message confirms the success of the process.
  - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
  - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
  - c. Click **Finish** to close the Installation Manager.

**Related information:**

 [Installation Manager updates](#)

*Installing interactively on an existing installation of WebSphere Application Server:*

Using custom installation, you can install IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
  - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```

- If you are installing from images downloaded from Passport Advantage, perform the following steps:
  - a. Go to the directory into which you extracted the images.
  - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

**Note:** Ensure that you have read/write access to the folders specified in the bpm\_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Installation on an existing WebSphere Application Server**.
5. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
6. Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
7. Click **Install**.
8. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server.
9. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

**Important:** Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

10. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.

11. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.  
If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.
12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
  - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
  - b. Optional: Click a feature to view its brief description under **Details**.
  - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

**Restriction:** Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
14. When the installation process is complete, a message confirms the success of the process.
  - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
  - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
  - c. Click **Finish** to close the Installation Manager.

**Related information:**

 [Installation Manager updates](#)

*Installing IBM Business Process Manager silently:*

You can install the IBM Business Process Manager product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

**Related reference:**

 [IBM Business Process Manager Advanced system requirements](#)

*Installing silently using the command line:*

You can install IBM Business Process Manager silently using the command line. With this method, you only need to run one command to perform the installation.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

```
extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.

3. Run the following command:

```
extract_directory/IM/tools/imcl install list_of_product_IDs -acceptLicense
-installationDirectory location -repositories repository -showVerboseProgress
-log logName.log
```

where:

- *list\_of\_product\_IDs* is a list of the IDs for the products and features that you want to install. You must include the required features. The syntax is *productID,feature,feature*, with multiple products separated by spaces.

*Table 105. Product IDs*

| Product          | Product ID          | Feature and Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IBM BPM Advanced | com.ibm.bpm.ADV.V80 | <ul style="list-style-type: none"> <li>• <code>bpmAdv.nonprod</code>: Test, staging, or development use. You must specify either this feature or <code>bpmAdv.prod</code>.</li> <li>• <code>bpmAdv.prod</code>: Production use. You must specify either this feature or <code>bpmAdv.nonprod</code>.</li> <li>• <code>wps.client.feature</code>: Required. Run client applications that interact with a IBM BPM Advanced: Process Server in the same cell.</li> <li>• <code>wps.server.feature</code>: Required. Full installation of IBM BPM Advanced: Process Server, which includes the client code and the server code.</li> </ul> |

Table 105. Product IDs (continued)

| Product                                         | Product ID               | Feature and Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------------------------------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WebSphere Application Server Network Deployment | com.ibm.websphere.ND.v80 | <ul style="list-style-type: none"> <li>• core.feature: Required. WebSphere Application Server core content.</li> <li>• ejbdeploy: Required. Pre-Enterprise JavaBeans (EJB) 3.0 modules.</li> <li>• thinclient: Required. Stand-alone thin clients and resource adapters.</li> <li>• embeddablecontainer: Embeddable EJB container.</li> <li>• samples: Sample applications feature.</li> <li>• com.ibm.sdk.6_32bit: 32-bit software development kit (SDK). You must specify either this feature or com.ibm.sdk.6_64bit.</li> <li>• com.ibm.sdk.6_64bit: 64-bit SDK. This feature can only be selected on a 64-bit system. You must specify either this feature or com.ibm.sdk.6_32bit.</li> </ul> |
| Installation Manager                            | com.ibm.cic.agent        | <ul style="list-style-type: none"> <li>• agent_core: Installation Manager core content.</li> <li>• agent_jre: Installation Manager Java Runtime Environment (JRE).</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

- *location* is the path to the directory where you want to install the products.
- *repository* is the path to the repository where you have extracted the files, one of the following directories:

*extract\_directory/repository/repos\_32bit*  
*extract\_directory/repository/repos\_64bit*

For more than one repository, separate the repository locations with commas.

- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas. Create encrypted passwords using the IBM Installation Manager.

Table 106. Keys

| Key                   | Description                                                                                                                                                                           |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| user.db2.use.existing | <p>If you are using an existing DB2 setup, add the following line exactly:</p> <pre>user.db2.use.existing=true</pre> <p>If DB2 must be installed, specify the value <i>false</i>.</p> |
| user.db2.port         | Port for the DB2 database. The default value is 50000.                                                                                                                                |

- *logName* is the name of the log file to record messages and results.

Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for imcl.

Installation Manager installs the list of products and writes a log file to the directory that you specified. The log file is empty if there are no errors or warnings.



The following example installs IBM Business Process Manager Advanced and WebSphere Application Server Network Deployment on AIX.

```
imcl install com.ibm.bpm.ADV.v80,bpmAdv.prod,wps.client.feature,wps.server.feature
com.ibm.websphere.ND.v80,core.feature,ejbdeploy,thinclient,embeddablecontainer,samples,com.ibm.sdk.6_32bit
-acceptLicense -installationDirectory /usr/IBM/BPM80 -repositories /usr/tmp/BPM/repository
-showVerboseProgress -log silentinstall.log
```

**Important:** Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

**Related reference:**

[➦](#) Command-line arguments for imcl

[➦](#) IBM Business Process Manager Advanced system requirements

**Related information:**

[➦](#) IBM WebSphere Application Server Information Center

*Installing silently using a response file:*

You can install IBM Business Process Manager Advanced silently by creating a response file and then running a command to use that response file to install the product.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

By using response files, you can simplify the silent installation and reduce error in the process because you set up your installation options once in a saved, sharable file that can be used on one or more machines. The installation software provides sample response files for each supported operating system and bit version. You can use an unmodified sample response file to perform a silent installation using default settings, or you can modify the response file to set particular values. The comments in the sample response files provide detailed instructions and information about setting the values.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.  
`extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt`
2. Create the response file that will install the required base products and IBM Business Process Manager Advanced. Copy a sample response file, suitable for your bit version and user access level, from the following directory:

```
extract_directory/responsefiles/BPM/
```



Alternatively, you can create a response file by recording your actions in Installation Manager. When you record a response file, the selections that you make in Installation Manager are stored in an XML file. When you run Installation Manager in silent mode, Installation Manager uses the data in the XML response file to perform the installation.

3. The default values provided in the sample response files will perform a basic install, but you should review the file and its comments, and modify the parameters as needed for your environment and the access level of your user ID.

In particular, review the following parameters and values:

- For non-root user IDs, check that all location variables point to locations that the user has permissions for.
  - Repository location. If you are not running directly from the *extract\_directory/responsefiles/BPM/* directory, point to the location of your installation repository. The repository can be local or remote. If you are running from the DVD, copy the response file from the DVD and point it back at the repository on the DVD.
  - Installation location (**installLocation**) where IBM Installation Manager is already installed or will be installed.
  - Installation location where IBM Business Process Manager Advanced will be installed.
  - Eclipse location (**eclipseLocation**).
  - The list of features for the product.
  - Production or non-production use.
4. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.
  5. Run the following command:

Root user:

```
extract_directory/IM/installlc -acceptLicense input
extract_directory/responsefiles/BPM/response_file_name.xml
-log preferred_log_location/silent_install.log
```

Nonroot user:


```
extract_directory/IM/userinstc -acceptLicense input
extract_directory/responsefiles/BPM/response_file_name.xml
-log preferred_log_location/silent_install.log
```

Installation Manager installs any required prerequisites and IBM Business Process Manager Advanced, and writes a log file to the directory you specified.

**Important:** Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

**Related tasks:**

 [Installing silently by using Installation Manager](#)

 [Recording a response file with Installation Manager](#)

**Related reference:**

 [IBM Business Process Manager Advanced system requirements](#)

*Granting write permission of files and directories to nonroot users for profile creation or augmentation:*

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install\_root/profiles* directory of the product.

#### **Restrictions:**

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
  - a. Create a group named **profilers**, which will contain all users who can create profiles.
  - b. Create a user named **user1**, who can create profiles.
  - c. Add users **product\_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
  - Create the *install\_root/logs/manageprofiles* directory:  
`mkdir install_root/logs/manageprofiles`
  - Create the *install\_root/properties/fsdb* directory:  
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the *profileRegistry.xml* file as the installer. For this example, the file path is:  
`install_root/properties/profileRegistry.xml`

Add the following information to the *profileRegistry.xml* file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **/opt/IBM/WebSphere/AppServer**.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

The root user and the nonroot user can use the same tasks to manage profiles.

*Configuring profiles and databases:*

After you install the product, if you did not use typical installation, you must create or augment one or more profiles to define the runtime environment. Before starting a profile, you must have configured the databases that are to be used with the profile.

*Configuring a stand-alone environment with a DB2 database server on AIX:*

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 database server.

*Creating DB2 databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECON 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run `vi /etc/profile`

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run `vi /etc/group`

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102;root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

*Creating database design files for DB2:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

## Creating database design files for stand-alone profiles with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5)wesb.nd.topology
(6)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2)bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:



[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
```



```
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

-----

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
```

```

(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibME] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibME] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]

```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

```

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]

```

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.standalone is complete with 0 remaining item(s):
```

- When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

- Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.

- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with DB2 in a stand-alone environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).
```

`-v db_design_file | db_scripts_output_directory`  
when a `db_design_file` is given, validation will be done on the specified database design file based on the database specs.  
When a `db_scripts_output_directory` is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

`-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]`  
generate the database scripts from the specified design files in batch mode.  
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Database User name[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

Regular pagesize[default=32k] :

Regular TableSpace[default=TWSSSPACE] :

Temporary pagesize[default=32k] :

Temporary TableSpace[default=TWSTEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.



## Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

## Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

## Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating or augmenting stand-alone profiles with a DB2 database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Preparing the DB2 database server:*

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

If you plan to configure the databases on a remote server, you must have created the databases before beginning to create or augment the profile. You can create databases on the local server or use existing databases on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run  
`vi /etc/profile`

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run  
`vi /etc/group`



and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102;root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:  
`/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:  
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/  
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")`
- When the DB2 database manager is not running:  
`SQL1032N No start database manager command was issued. SQLSTATE=57019`
- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:  
`SQL1092N "ROOT" does not have the authority to perform the requested command.`

*Creating Process Center stand-alone profiles with a DB2 database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a web server definition.
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.

- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 107. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.

Table 107. Required database configuration fields for DB2 Database (continued)

Field	Action needed
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

**Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating Process Server stand-alone profiles with a DB2 database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.



The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a web server definition.
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.



- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Optional: Advanced:** To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**.
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.

- Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. Advanced: Configure the databases using a design file.

- Select **Use a database design file**.
- Click **Browse**.
- Specify the fully qualified path name for the design file.
- To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.

- From the **Select a database product** list, select **DB2**.
- Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
- Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
- To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
- Click **Next**.

- On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 108. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

- Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
- Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
- Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

- On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
- On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
- default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
- default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
- default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
- default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.

2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_Standalone\_DB2.response
- PC\_Std\_Standalone\_DB2z0S.response
- PC\_Std\_Standalone\_Oracle.response

- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

*Sample response file for an Advanced Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
```



```

If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development

```



```

standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative

```

```

security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to

```

```

set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck

```

```

#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false

```

```

true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2

```

```
dbHostName=db_host_name
dbServerPort=db_port_number
```

```
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
```

```
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
```

```
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
Valid Values:
Varies
Default Values:
None
#
```

```
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
Valid Values:
Varies
Default Values:
None
```

```
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
```



```

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserId
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: cdbSchemaName
#
Description:
The schema name for the Common database. If no value is specified, the
following default database schema names are used:

```

```

DB2 database
The default schema name is the value specified by the dbUserId parameter.
The -cdbSchemaName parameter is valid for all database types except for
Oracle and Microsoft SQL Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

```

```

#####
Parameter: dbCeiMeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter

```

```

has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter

```

```

is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFoamsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.

```

```

It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
true
Default Values:
true
#####
configureBPC=true

```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

```



```

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None

```

```

#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#

```

```

Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false

```

```

true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify

```

```

false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####

```

dbCreateNew=true

#####

# Parameter: dbType

#

# Description:

# The database type. Set one of the following values for the type of database product you are using with IBM Business Process Manager.

#

# Required only for one of the following:

# A stand-alone profile when -bpmdbDesign was not specified.

# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# DB2\_DATASERVER

# Default Values:

# DB2\_DATASERVER

#

# Parameter: dbJDBCClasspath

#

# Description:

# The directory path location of JDBC driver files.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbHostName

#

# Description:

# The database server host name or IP address. The default value is the local host name. Required when the -bpmdbDesign parameter was not specified. (It is not required if the -bpmdbDesign parameter is specified.)

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbServerPort

#

# Description:

# The database server port number. Depending on the database you are using, you can specify a different port number instead of the default port number.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbType=DB2\_DATASERVER

dbJDBCClasspath=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/DB2

dbHostName=db\_host\_name

dbServerPort=db\_port\_number

#####



```

Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#

```

```

Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

```

```

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserID
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: cdbSchemaName
#
Description:
The schema name for the Common database. If no value is specified, the
following default database schema names are used:
DB2 database
The default schema name is the value specified by the dbUserId parameter.
The -cdbSchemaName parameter is valid for all database types except for
Oracle and Microsoft SQL Server.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a

```

```

preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#

```

```

was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName

```



```

#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#

```

```

Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:

```

```

false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType

```

```

#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is

```

```

created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#

```



```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#

```

```

Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password
#####
Parameter: cdbSchemaName
#
Description:
The schema name for the Common database. If no value is specified, the
following default database schema names are used:
DB2 database
The default schema name is the value specified by the dbUserId parameter.
The -cdbSchemaName parameter is valid for all database types except for
Oracle and Microsoft SQL Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbAppMeUserId=app_me_userid
#dbAppMePassword=app_me_pwd
#####
Parameter: dbCeimeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

```

```

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:

```

```

false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false
#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure

```

```

Business Process Choreographer.
#
Valid Values:
false
true
Default Values:
true
#####
configureBPC=true

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that

```



```

you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsrv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each

```

```

profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:

```

```

Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the

```

```

root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port

```

```

settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:

```

```

localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID

```

```

#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```



```

Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

```

```

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: cdbSchemaName
#
Description:
The schema name for the Common database. If no value is specified, the
following default database schema names are used:
DB2 database
The default schema name is the value specified by the dbUserId parameter.
The -cdbSchemaName parameter is valid for all database types except for
Oracle and Microsoft SQL Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false

```

```
true
Default Values:
false
#####
#isDeveloperServer=false
```

*Augmenting stand-alone profiles for Process Center with a DB2 database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a web server definition.
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.
 

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
8. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 109. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/db2/java.



Table 109. Required database configuration fields for DB2 Database (continued)

Field	Action needed
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

**Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles for Process Server with a DB2 database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a web server definition.
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command **install\_root/bin/ProfileManagement/pmt.sh**.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
8. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name**: The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type**: Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol**: Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name**: Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port**: Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name**: Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password**: Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **DB2**.

- b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 110. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/db2/java.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

- 12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
- 13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
- 14. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

- 15. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
- 16. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:
  - `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
  - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
  - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2z0S.response



- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Configuring DB2 databases after stand-alone profile creation:*

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*Creating the Process Server database tables:*

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root/dbscripts/ProcessServer/DB2/* at profile creation time.
2. Run the following script.  
`configProcessServerDB.sh`

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

*Creating the Performance Data Warehouse database tables:*

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Chosen not to run the database scripts to initialize the databases during the configuration.
  - Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root/dbscripts/PerformanceDW/DB2/* at profile creation time.
  2. Run the following script.  
`configPerformanceDWDB.sh`

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

*Creating the Common database tables:*

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
  - Chosen not to run the database scripts to initialize the Common database during the configuration.
  - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install\_root/profile/dbscripts/CommonDB/DB2/*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db\_type* represents the supported database product and *db\_name* represents the name of the database.



You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

**configCommonDB.sh createDB** - creates the database and tables

**configCommonDB.sh** - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase\_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable\_AppScheduler.sql**
- c. **createTable\_CommonDB.sql**
- d. **createTable\_customization.sql**
- e. **createTable\_DirectDeploy.sql**
- f. **createTable\_EsbLoggerMediation.sql**
- g. **createTable\_governancerepository.sql**
- h. **createTable\_lockmanager.sql**
- i. **createTable\_mediation.sql**
- j. **createTable\_Recovery.sql**
- k. **createTable\_RelationshipService.sql**
- l. **insertTable\_CommonDB.sql**

*Creating the Business Space database tables:*

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install\_root/profile/dbscripts/BusinessSpace/node\_name\_server\_name/DB2/CMNDB*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:  
**configBusinessSpaceDB.sh** Connect to the database, bind the Command Line Interface to the Business Space database using the **db2 bind DB2\_installation\_directory\bnd\@db2cli.lst blocking all grant public** command, and then reset the database connection.

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.timestamp.log`. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

*Configuring a stand-alone environment with a DB2 for z/OS database server on AIX:*

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

*Creating database design files and database scripts (AIX stand-alone):*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS:*

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.

- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5)wesb.nd.topology
(6)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2)bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
```

```
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database Location on z/OS system[default=] :
```

```
Database server host[default=] :
```

```
Database server port[default=] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```



[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

**Note:** For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM\_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1)false
- (2>true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.standalone**, after you configure the master component, you must manually configure the **BPM\_PerformanceDW** and **BPM\_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.



[info] Please pick one of the following [database component(s)] :

- (1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2) [BPC] WBI\_BPC : [status = complete]
- (3) [BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4) [BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI\_BSPACE : [status = complete]
- (6) [SibME] WBI\_BPC\_ME : [status = complete]
- (7) [SibME] WBI\_CEI\_ME : [status = complete]
- (8) [SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9) [SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10) [SibME] BPM\_PerformanceDW\_ME : [status = complete]
- (11) [SibME] BPM\_ProcessServer\_ME : [status = complete]
- (12) [save and exit]

Please enter the number for the database component :12

[status] bpm.advanced.standalone.topology is not complete with 6 remaining item(s):

- [ 1 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'schemaName' for SCHEMA is empty.
- [ 2 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'catalog' for VCAT is empty.
- [ 3 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'adminUserName' for SECURITY\_ADMIN\_USER is empty.
- [ 4 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'adminPassword' for SECURITY\_ADMIN\_PASSWORD is empty.
- [ 5 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects : required property 'schemaName' for SCHEMA is empty.
- [ 6 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects : required property 'catalog' for VCAT is empty.

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

[info] The script(s) have been generated in  
WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI\_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.

- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment:*

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
```

scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.  
Database name[default=BPMDB] :  
Data source user name[default=] :  
Database schema name (SQLID)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :  
The user ID you use for administrative security[default=] :  
The password for the name specified with the adminUserName parameter[default=] :  
Bufferpool of 4K size for tables[default=BP1] :  
Bufferpool of 4K size for LOB data[default=BP3] :  
Bufferpool of 8K size for tables[default=BP8K1] :  
Bufferpool of 16K size for tables[default=BP16K1] :  
Bufferpool of 32K size for tables[default=BP32K1] :  
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- 8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:  

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```
- 9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:  

```
generate database scripts? (y/n) [default=y] :
```
- 10. To generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:  

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

#### Related tasks:

“Creating the messaging engine data stores” on page 1207

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

### *Generating scripts for creating database objects in DB2 for z/OS:*

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
  - Database server details
  - Data source details
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.



For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.

2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: cd /opt/IBM/WebSphere/AppServer/util/dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db\_design\_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

**Tip:** In the database design file, the directories for the SQL scripts are defined by the *ddl\_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl\_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.  
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

**Note:** The **createDB2.sh** script requires execute permissions.



*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### **Required property is empty errors**

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### **Sample output of running a validation of the existing database design**

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### **Contents of the database design tool log file**

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating or augmenting stand-alone profiles with a DB2 for z/OS database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Creating Process Center stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Run the command ***install\_root*/bin/ProfileManagement/pmt.sh**.

2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:

- In the **Path** field, enter the directory path to the existing certificate.
- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. **Advanced:** On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating Process Server stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Run the command *install\_root*/bin/ProfileManagement/pmt.sh.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root*/profiles/*profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.



9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

- Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - Select **Create a web server definition**.
  - Specify the web server characteristics on the page, and click **Next**
  - Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - Click **Next**.
- On the Process Server Configuration page, set the values for the following parameters.
  - Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.



- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

- Advanced: Configure the databases using a design file.
  - Select **Use a database design file**.
  - Click **Browse**.
  - Specify the fully qualified path name for the design file.
 

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - Click **Next**. The design file location is passed to the command line to complete the database configuration.
- Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
- Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
- On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
- On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

- Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
  - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
  - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
- Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
- To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response

- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

*Sample response file for an Advanced Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
```

```

and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a

```

```

unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.

```

```

procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an

```



```

automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#

```

```

Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```

```

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbidbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false

```

```

true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:

```

```

Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFoMServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

```



```
#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false
```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses DB2z0S.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2z0S.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
```

```

Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####

```

```

Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#

```

```

Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:

```

```

false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#

```



```

Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#

```

```

Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId

```

```

#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or

```

```

test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

```

```

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the

```

```

file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else

```

```

shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity

```



```

#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.

```

```

During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.

```

```

#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database

```

```

authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

```

```

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies

```



```

Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.

```

```

Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94

```

```

(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:

```

```

profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.

```

```

This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period

```

```

#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or

```



```

false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword

```

```

#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

```

```

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB

```

```

#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####

```

```

Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####

Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####

Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false
#####

```

*Augmenting stand-alone profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Run the command **install\_root/bin/ProfileManagement/pmt.sh**.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. Advanced: Configure the databases using a design file.
    - a. Select **Use a database design file**.
    - b. Click **Browse**.
    - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.

- d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
9. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
10. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
11. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
12. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Augmenting stand-alone profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Run the command **install\_root/bin/ProfileManagement/pmt.sh**.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application



Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.
    - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
    - **Environment Type:** Select how the Process Server is used:
      - Select **Production** if the server is to be used in a production capacity.
      - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
      - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

9. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.

- d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
10. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
11. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
12. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
13. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:
  - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
  - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
  - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions

are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response

- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating and configuring DB2 for z/OS databases after stand-alone profile creation:*

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*DB2 for z/OS authorization prerequisites:*

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

## User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as #DB\_USER#. These permissions are available in the createDatabase.sql database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the #DB\_USER# user to create sequences and stored procedures with a schema qualifier of #SCHEMA#:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

## Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database #DB\_NAME# can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
```

```

GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;

```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named S1CELL, and connect user WSADMIN to it, as follows:

```

INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
WSADMIN CONNECT 000000 NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE RESUME DATE=NONE

```

## Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

## Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.



## Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

### Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Creating databases in the DB2 for z/OS subsystem:*

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

### Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

### Considerations for choosing the createDB2.sh script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

### Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two .sql files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

### Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Configuring the DB2 command line processor:*

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
  - Modify the CLASSPATH environment variable to include the `clp.jar` file.
  - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
  - Define the **db2** command as an alias for the command that starts the command line processor.
  - Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:



```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=
/file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=
/wasv8config/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:
  - A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
  - A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

**Tip:** When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

#### Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script”

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 1203

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (*z\_schema.sql* and *z\_schemaProc.sql*) that you can run later by using the DB2 command line processor.

*Creating DB2 for z/OS database objects using the createDB2.sh script:*

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**-RunSQL**

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

**Tip:** The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 1200

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using the DB2 command line processor:*

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCreate
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**Tip:** By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two `.sql` files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

The `z_schema.sql` and `z_schemaProc.sql` files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The `z_schemaProc.sql` file contains SQL statements for stored procedures, and is generated in addition to `z_schema.sql` because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the `z_schema.sql` file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the `z_schemaProc.sql` file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 1200

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:*

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.

- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

**Tip:** The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

**SPUFI** A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

#### DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:  

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f IS08859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

**Important:** After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

**Tip:** If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t IS08859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:
  - @STOGRP@ = DB2 Storage group name
  - @DB\_NAME@ = DB2 Database name
  - @SCHEMA@ = DB2 SQLID (Schema Qualifier)
  - @BPTABLE4K@ = Buffer Pool of 4k Size
  - @BPINDEX@ = Buffer Pool Index
  - @VOLUMES@ = DB2 comma separated list of volume-ids
  - @VCAT@ = DB2 integrated catalog facility name
  - @DB\_USER@ = The user ID which will be used to create the database objects
4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.



**Tip:** If you previously ran the `createDB2.sh` script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (`z_schema.sql` and `z_schemaProc.sql`), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

**Related tasks:**

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 1203

When you run the `createDB2.sh` script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the `createDB2.sh` script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

*Creating the messaging engine data stores:*

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
  - When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
    - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
    - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
    - a. Run the database design tool (**DbDesignGenerator**) from the `install_root/util/dbUtils` directory.
    - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
    - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

**Tip:** As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the

tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

**Related tasks:**

“Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment” on page 1130

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS” on page 1123

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

*Granting table privileges to the JCA authentication alias user ID:*

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

**Note:** Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.



**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

*Setting the correct schema name for the messaging engines:*

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
  - a. Select **Messaging engines**, and then click the name that is displayed.
  - b. Click **Message store**.
  - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
  - d. Click **Apply**.
  - e. Save your configuration changes.
4. Verify that the messaging engines can be started:
  - a. Log out of the administrative console.
  - b. Stop, and then restart the server.
  - c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

*Configuring a stand-alone environment with an Oracle database server on AIX:*

You can configure a stand-alone environment for IBM Business Process Manager using an Oracle database server.

### *Setting up Oracle databases:*

You or your database administrator (DBA) can configure the Process Server database, Performance Data Warehouse database, and Common database before, after, or during creating or augmenting a profile.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

You have the following options for creating and configuring the databases:

- You or your DBA creates the databases in advance and the Profile Management Tool configures them for you during profile creation or augmentation.
- You or your DBA creates a database design file using the database design tool, and you pass the file to the Profile Management Tool or **manageprofiles** command-line utility during profile creation or augmentation.
- You postpone the database configuration, and you or your DBA creates the required databases and database tables later by running the scripts that are generated during profile creation or augmentation.

### *Creating database design files for Oracle:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Creating database design files for stand-alone profiles with Oracle:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
```

- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2) bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.  
 [info] Completed database components can be edited to change existing or defaulted property values.  
 [info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI\_BPC : [status = not complete]
- (3) [BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4) [BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI\_BSPACE : [status = not complete]
- (6) [SibME] WBI\_BPC\_ME : [status = not complete]
- (7) [SibME] WBI\_CEI\_ME : [status = not complete]
- (8) [SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9) [SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10) [SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11) [SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI\_CommonDB is not complete with 1 remaining item(s):  
 [ 1 ] WBI\_CommonDB.WBI\_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool

is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

[info] Please pick one of the following [Oracle driver type(s)] :

- (1)oci8
- (2)thin

Please enter the number for the Oracle driver type(s)? [default=thin] :  
 Data source user name[default=] :  
 Data source password[default=] :  
 Oracle JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/Oracle  
 ] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI\_CommonDB is complete with 0 remaining item(s):

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.standalone**, you must manually configure the **BPM\_PerformanceDW**, **BPM\_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]



```
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

Please enter the number for the database component :3

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[Bspace] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

Please enter the number for the database component :4

```

[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:



[info] The script(s) have been generated in  
WAS\_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI\_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with Oracle in a stand-alone environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
```

- (6)cei
- (7)sca
- (8)sibme
- (9>wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processerver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=BPMDB] :

Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
```

```
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
WAS_HOME/profiles/default/util/dbUtils/Oracle-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating or augmenting stand-alone profiles with an Oracle database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Preparing the Oracle database server:*

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

To configure the database information while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

*Creating Process Center stand-alone profiles with an Oracle database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a web server definition.
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.



5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page



When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.

- b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 111. Required database configuration fields for Oracle

Field	Action needed
<b>Common database</b>	<p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p>
<b>Process Server database</b>	<p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
<b>Performance Data Warehouse database</b>	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1521 or enter the correct server port number.
<b>Directory location of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating Process Server stand-alone profiles with an Oracle database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.

- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.

Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do



not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.



- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.

17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 112. Required database configuration fields for Oracle

Field	Action needed
<b>Common database</b>	<p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p>
<b>Process Server database</b>	<p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
<b>Performance Data Warehouse database</b>	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1521 or enter the correct server port number.
<b>Directory location of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

- `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
- `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
- `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
- `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
- `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.

2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response

- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
```

```

#####
templatePath=BPM/default.procctr.adv
#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01
#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```



```

#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:

```

```

The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:

```

```

manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#

```

```

Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####

```

```

Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```



```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword

```

```

#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pwd

#####
Parameter: dbAppMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd

#####
Parameter: dbCeiMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
user-specified a user name and password during profile creation, this
parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword. If you enter a user-specified a user name and password during
profile creation, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test

```

```

Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
```



```

Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```

```

Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is

```

```

localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:

```

```

manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For

```

```

databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle

```



```

databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

```

```

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
```

```

#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#

```



```

Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```

```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbdbDesign that was used in previous

```

```

versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

```

```

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:

```



```

Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pwd

#####
Parameter: dbAppMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd
#####
Parameter: dbCeiMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd
#####
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
user-specified a user name and password during profile creation, this
parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword. If you enter a user-specified a user name and password during
profile creation, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#

```

```

Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

```

```

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a

```

```

Standalone PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be

```

```

currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#

```

```

Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword

```



```

along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#

```

```

Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winServiceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceCheck
#
Description:

```

```

The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false

```

```

#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####

```

```

dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

```

```

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####

```

```
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
```

```
#####
```

```
Parameter: dbProcSvrMeUserId
```

```
#
```

```
Description:
```

```
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
```

```
#
```

```
Note: Do not use this parameter if you are using -bpmdbDesign.
```

```
#
```

```
Valid Values:
```

```
Varies
```

```
Default Values:
```

```
None
```

```
#
```

```
Parameter: dbProcSvrMePassword
```

```
#
```

```
Description:
```

```
Note: Do not use this parameter if you are using -bpmdbDesign.
```

```
#
```

```
Valid Values:
```

```
Varies
```

```
Default Values:
```

```
None
```

```
#####
```

```
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
```

```
dbProcSvrMePassword=db_ProcSvr_Me_Password
```

```
#####
```

```
Parameter: dbPerfDWMeUserId
```

```
#
```

```
Description:
```

```
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
```

```
#
```

```
Valid Values:
```

```
Varies
```

```
Default Values:
```

```
None
```

```
#
```

```
Parameter: dbPerfDWMePassword
```

```
#
```

```
Valid Values:
```

```
Varies
```

```
Default Values:
```

```
None
```

```
#####
```

```
dbPerfDWMeUserId=db_PerfDW_Me_UserId
```

```
dbPerfDWMePassword=db_PerfDW_ME_Password
```

```
#####
```

```
Parameter: dbCommonUserId
```

```
#
```

```
Description:
```

```
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
```



```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:

```

```

Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting stand-alone profiles for Process Center with an Oracle database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
  - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  8. Advanced: Configure the databases using a design file.
    - a. Select **Use a database design file**.
    - b. Click **Browse**.
    - c. Specify the fully qualified path name for the design file.
    - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.
- Skip to the Business Space Configuration step
9. On the Database Configuration - Part 1 page, specify your database information.
    - a. From the **Select a database product** list, select **Oracle**.
    - b. Optional: If you want to change the default values, enter new names for the required databases.
      - The database names can be the same.
      - You must input the database service name for each database.

- The database must already exist.
- c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 113. Required database configuration fields for Oracle

Field	Action needed
<b>Common database</b>	<p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p>
<b>Process Server database</b>	<p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>

Table 113. Required database configuration fields for Oracle (continued)

Field	Action needed
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  13. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
  14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles for Process Server with an Oracle database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command **install\_root/bin/ProfileManagement/pmt.sh**.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.
    - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
    - **Environment Type:** Select how the Process Server is used:
      - Select **Production** if the server is to be used in a production capacity.
      - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
      - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.



- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.
- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.



Table 114. Required database configuration fields for Oracle

Field	Action needed
<b>Common database</b>	<p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p>
<b>Process Server database</b>	<p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
<b>Performance Data Warehouse database</b>	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b> Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b> Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b> Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1521 or enter the correct server port number.
<b>Directory location of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
  15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
 Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_Standalone\_DB2.response
- PC\_Std\_Standalone\_DB2zOS.response
- PC\_Std\_Standalone\_Oracle.response
- PC\_Std\_Standalone\_SQLServer.response
- PC\_Adv\_Standalone\_DB2.response
- PC\_Adv\_Standalone\_DB2zOS.response
- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response

- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the

**bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Configuring Oracle databases after stand-alone profile creation:*

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*Creating the Process Server database tables:*

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root*/dbscripts/ProcessServer/Oracle/ at profile creation time.

2. Run the following script.

```
configProcessServerDB.sh
```

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

*Creating the Performance Data Warehouse database tables:*

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root*/dbscripts/PerformanceDW/Oracle/ at profile creation time.

2. Run the following script.

```
configPerformanceDWDB.sh
```

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

### *Creating the Common database tables:*

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
  - Chosen not to run the database scripts to initialize the Common database during the configuration.
  - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install\_root/profile/dbscripts/CommonDB/Oracle/*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db\_type* represents the supported database product and *db\_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run the following command:

**configCommonDB.sh** Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase\_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable\_AppScheduler.sql**
- c. **createTable\_CommonDB.sql**
- d. **createTable\_customization.sql**
- e. **createTable\_DirectDeploy.sql**
- f. **createTable\_EsbLoggerMediation.sql**
- g. **createTable\_governancerepository.sql**
- h. **createTable\_lockmanager.sql**
- i. **createTable\_mediation.sql**
- j. **createTable\_Recovery.sql**
- k. **createTable\_RelationshipService.sql**
- l. **insertTable\_CommonDB.sql**

### *Creating the Business Space database tables:*

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install\_root/profile/dbscripts/BusinessSpace/node\_name\_server\_name/Oracle/CMNDB*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:

**configBusinessSpaceDB.sh** Alternatively, you could customize and run the individual scripts manually in the following order:

- a. **createDatabase.sql**
- b. **createTable\_BusinessSpace.sql**
- c. **createGrant\_BusinessSpace.sql**

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

*Configuring a stand-alone environment with an SQL Server database server on AIX:*

You can configure a stand-alone environment for IBM Business Process Manager using a Microsoft SQL Server database server.

*Creating SQL Server databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.



The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
  - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db\_user\_account** and **db\_user\_password** are the user name and password that are used to log in to create the database, and **database\_name** is the name of the database that you are creating. **COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

**Important:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default



schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

#### *Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

#### **On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

#### **On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
  - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
  - c. Right-click **Local DTC** and then select **Properties**.
  - d. Click the **Security** tab on the **Local DTC Properties** window.
  - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
  - f. Click **OK** again to close the Properties window, and then close Component Services.
  - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
    - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
    - b. Extract the archive to any folder.

- c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the x64 folder.
- d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

*Creating database design files for SQL Server:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for stand-alone profiles with SQL Server:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.
```

```

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.

```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]

```

Please enter the number for the design option :

- To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

- To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2)bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

- Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

- To edit the database configuration for the component, type **y** and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

- Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database server host[default=] :
Database User name[default=] :
Database schema[default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :
```

```
[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :
```

- (1) false
- (2) true

```
Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the **BPM\_ProcessServer**, you must manually configure the **BPM\_PerformanceDW** component for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(5)[save and exit]
```

Please enter the number for the database component :2

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:



[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with SQL Server in a stand-alone environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/uti/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/uti/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:



[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm\_performancedw
- (4) bpm\_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 4 to select **(4) SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.  
Database name[default=BPMDDB] :  
Database server host[default=] :  
Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1)false  
(2>true

Please enter the number for the Is this database for a Process Center? [default=false] :  
The user ID you use for administrative security[default=] :  
The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.  
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

(1) false  
(2) true

Data source user name[default=] :  
Data source password[default=] :  
MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating or augmenting stand-alone profiles with an SQL Server database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Preparing the SQL Server database server:*

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

**Restriction:** The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

*Table 115.*

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ\_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner dbo using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDBE00 (Business Process Choreographer Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)

- MEDPE00 (Performance Data Warehouse messaging engine)

IBM Business Process Manager packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the Detailed hardware and software requirements for IBM Business Process Manager page.

**Note:** You are responsible for providing JDBC driver levels outside of what is packaged with IBM Business Process Manager.

You must configure XA transactions after the database is installed and before you start the server.

*Creating Process Center stand-alone profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command **install\_root/bin/ProfileManagement/pmt.sh**.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next** .
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page



When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.

- b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 116. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the <b>Apply Windows Authentication</b> option.	
<b>Common database</b>	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Common database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Common database.</li> </ul>



Table 116. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Process Server database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Process Server database.</li> </ul>
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 117. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul>	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
  - **Windows Authentication:** With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
17. **Advanced:** On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  18. **Advanced:** Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### **Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating Process Server stand-alone profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root/profiles/profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for

commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.

- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Optional: Advanced:** To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**.
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.



- Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
- Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. **Advanced:** Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.

17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 118. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the <b>Apply Windows Authentication</b> option.	
<b>Common database</b>	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Common database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Common database.</li> </ul>
<b>Process Server database</b>	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Process Server database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Process Server database.</li> </ul>
<b>Performance Data Warehouse database</b>	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.
<b>Directory location of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 119. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul>	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.



19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 

Templates for each profile are located in the *install\_root/profileTemplates/BPM* directory for BPM templates and under *install\_root/profileTemplates* for other products. The following templates are available:

  - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
  - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
  - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- **PC\_Std\_DMGr\_DB2.response**
- **PC\_Std\_DMGr\_DB2zOS.response**
- **PC\_Std\_DMGr\_Oracle.response**
- **PC\_Std\_DMGr\_SQLServer.response**
- **PC\_Adv\_DMGr\_DB2.response**

- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a stand-alone profile that includes the CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space components in one database.

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates/BPM/
default.procsvr.adv -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath
path to JDBC driver -dbUserId userID -dbPassword password -dbCommonForME true -dbCeimeSchemaName
ceiSchema -dbSysMeSchemaName sysSchema
-dbAppMeSchemaName appSchema
```

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

*Sample response file for an Advanced Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
```

```

(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:

```

```

profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.

```

```

This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid

```

```

for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts

```



```

parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#

```

```

Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false

```

```

#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCclasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####

```

```

dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

```

```

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: dbAppMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
Parameter: dbCeimeSchemaName
#
Valid Values:
Varies

```



```

Default Values:
None
#
Parameter: dbCeiMeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd
#####
Parameter: dbSysMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
dbSysMeSchemaName=sys_me_schemaName
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

```

```

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

```

```

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:

```

```

WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01
#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber

```

```

where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```



```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbdbDesign that was used in previous

```

```

versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by

```

```

default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#
Description:

```

```

Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword

```



```

#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.

```

```

Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT
#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_SQLServer.response

```

```

#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####

```

profileName=ProcSrv01

```

Parameter: applyPerfTuningSetting

Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.

Valid Values:
development
standard
production
Default Values:
None

#applyPerfTuningSetting=standard

Parameter: cellName

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: nodeName

Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: serverName
#
```

```

Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####

```

```
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
```

```

#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName

```

```

#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false

```



```

true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:

```

```

The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.

```

```

Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name

```

dbProcSvrPassword=db\_ProcSvr\_Password

#####

# Parameter: perfDWDbName

#

# Description:

# Database name for Performance Data Warehouse. The value specified for  
# -perfDWDbName and -procSvrDbName must be different.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Parameter: dbPerfDWUserId

#

# Description:

# Performance Data Warehouse user name. Required for setting up the  
# Performance Data Warehouse.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Parameter: dbPerfDWPASSWORD

#

# Description:

# Performance Data Warehouse password. Required for setting up the  
# Performance Data Warehouse.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#####

perfDWDbName=perfDW\_Db\_Name

dbPerfDWUserId=db\_PerfDW\_UserId

dbPerfDWPASSWORD=db\_PerfDW\_Password

#####

# Parameter: dbProcSvrMeUserId

#

# Description:

# Process Server Messaging Engine user name. Required for setting up the  
# Process Server Messaging Engine. If you enter a user-specified a user name  
# and password during profile creation and if you specified ORACLE for  
# dbType, this parameter is required.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbProcSvrMePassword

#

# Description:

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbProcSvrMeUserId=db\_ProcSvr\_ME\_UserID

dbProcSvrMePassword=db\_ProcSvr\_Me\_Password

#####

# Parameter: dbPerfDWMeUserId

#

# Description:

# Performance Data Warehouse Messaging Engine user name. Required for  
# setting up the Performance Data Warehouse Messaging Engine.

#

```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: dbAppMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
Parameter: dbCeiMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeSchemaName=sys_me_schemaName
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:

```

```

Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#

```



```

Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false
#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false

```

```
Default Values:
false
#####
configureBPC=false
```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
```

```

path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber

```

```

else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the

```

```

-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:

```

```

Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####

```



```

Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

```

```

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName

```

```

#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId

```

dbPerfDWPassword=db\_PerfDW\_Password

#####

# Parameter: dbProcSvrMeUserId

#

# Description:

# Process Server Messaging Engine user name. Required for setting up the  
# Process Server Messaging Engine. If you enter a user-specified a user name  
# and password during profile creation and if you specified ORACLE for  
# dbType, this parameter is required.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbProcSvrMePassword

#

# Description:

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbProcSvrMeUserId=db\_ProcSvr\_ME\_UserId

dbProcSvrMePassword=db\_ProcSvr\_Me\_Password

#####

# Parameter: dbPerfDWMeUserId

#

# Description:

# Performance Data Warehouse Messaging Engine user name. Required for  
# setting up the Performance Data Warehouse Messaging Engine.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbPerfDWMePassword

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbPerfDWMeUserId=db\_PerfDW\_Me\_UserId

dbPerfDWMePassword=db\_PerfDW\_ME\_Password

#####

# Parameter: dbCommonUserId

#

# Description:

# The user id to authenticate with the Common database. For Oracle, the  
# default user name is the first three characters of the Oracle database  
# service name. For example: If the database service name is ORCL,  
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business  
# Process Manager profiles and advanced deployment manager (dmgr.\*.adv)  
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT\_SQLSERVER  
# and no bpmdbDesign file was specified.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pwd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:

```

```

Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting stand-alone profiles for Process Center with an SQL Server database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  8. Advanced: Configure the databases using a design file.
    - a. Select **Use a database design file**.
    - b. Click **Browse**.
    - c. Specify the fully qualified path name for the design file.
    - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 120. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the <b>Apply Windows Authentication</b> option.	
<b>Common database</b>	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Common database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Common database.</li> </ul>
<b>Process Server database</b>	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Process Server database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Process Server database.</li> </ul>



Table 120. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 121. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul>	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
  - **Windows Authentication:** With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
11. **Advanced:** On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  12. **Advanced:** Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  13. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### **Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles for Process Server with an SQL Server database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Run the command *install\_root/bin/ProfileManagement/pmt.sh*.
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.

- **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
- **Environment Type:** Select how the Process Server is used:
  - Select **Production** if the server is to be used in a production capacity.
  - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
  - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

- Advanced: Configure the databases using a design file.
  - Select **Use a database design file**.
  - Click **Browse**.
  - Specify the fully qualified path name for the design file.
  - To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

- On the Database Configuration - Part 1 page, specify your database information.
  - From the **Select a database product** list, select **Microsoft SQL Server**.
  - Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

d. Click **Next**.

11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 122. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the <b>Apply Windows Authentication</b> option.	
<b>Common database</b>	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Common database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Common database.</li> </ul>
<b>Process Server database</b>	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Process Server database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Process Server database.</li> </ul>
<b>Performance Data Warehouse database</b>	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b> Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b> Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b> Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.
<b>Directory location of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 123. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul>	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.



13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
 Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:
  - `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
  - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
  - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response



- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Configuring SQL Server databases after stand-alone profile creation:*

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*Creating the Process Server database tables:*

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root/dbscripts/ProcessServer/SQLServer/* at profile creation time.
2. Run the following script.  
`configProcessServerDB.sh`

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

*Creating the Performance Data Warehouse database tables:*

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Chosen not to run the database scripts to initialize the databases during the configuration.
  - Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root/dbscripts/PerformanceDW/SQLServer/* at profile creation time.
  2. Run the following script.  
`configPerformanceDWDB.sh`

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

*Creating the Common database tables:*

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the Common database during the configuration.
- Created the database that you want to create the Common database tables in.

1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install\_root/profile/dbscripts/CommonDB/SQLServer/*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db\_type* represents the supported database product and *db\_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

**configCommonDB.sh createDB** - creates the database and tables

**configCommonDB.sh** - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase\_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable\_AppScheduler.sql**
- c. **createTable\_CommonDB.sql**
- d. **createTable\_customization.sql**
- e. **createTable\_DirectDeploy.sql**
- f. **createTable\_EsbLoggerMediation.sql**
- g. **createTable\_governancerepository.sql**
- h. **createTable\_lockmanager.sql**
- i. **createTable\_mediation.sql**
- j. **createTable\_Recovery.sql**
- k. **createTable\_RelationshipService.sql**
- l. **insertTable\_CommonDB.sql**

*Creating the Business Space database tables:*

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.

1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install\_root/profile/dbscripts/BusinessSpace/node\_name\_server\_name/SQLServer/CMNDB*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:

**configBusinessSpaceDB.sh** Alternatively, you could customize and run the individual scripts manually in the following order:

- a. **createDatabase.sql**
- b. **createTable\_BusinessSpace.sql**
- c. **createGrant\_BusinessSpace.sql**

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

**Important:** If the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
```

```
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

### **Modifying an existing installation:**

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

### **Configuring a network deployment environment on AIX**

For a network deployment configuration, you install the software and then create or augment a deployment manager, create or augment at least one custom profile, and configure the network deployment environment.

### **Granting write permission of files and directories to nonroot users for profile creation or augmentation:**

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install\_root/profiles* directory of the product.

### **Restrictions:**

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
  - a. Create a group named **profilers**, which will contain all users who can create profiles.
  - b. Create a user named **user1**, who can create profiles.
  - c. Add users **product\_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
  - Create the *install\_root/logs/manageprofiles* directory:  
`mkdir install_root/logs/manageprofiles`
  - Create the *install\_root/properties/fsdb* directory:  
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the `profileRegistry.xml` file as the installer. For this example, the file path is:  
`install_root/properties/profileRegistry.xml`

Add the following information to the `profileRegistry.xml` file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **/opt/IBM/WebSphere/AppServer**.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.



The root user and the nonroot user can use the same tasks to manage profiles.

### Creating databases and database design files:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. You must create the databases before network deployment.

*Creating databases and database design files for DB2:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

*Creating DB2 databases:*

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the /etc/profile file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root user to the /etc/group administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103:db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:  
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:  
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
- When the DB2 database manager is not running:  
SQL1032N No start database manager command was issued. SQLSTATE=57019
- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:  
SQL1092N "ROOT" does not have the authority to perform the requested command.

*Creating database design files for DB2:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for deployment environments with DB2:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).
```



-v db\_design\_file | db\_scripts\_output\_directory  
 when a db\_design\_file is given, validation will be done on the specified database design file based on the database specs.  
 When a db\_scripts\_output\_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db\_design\_file [-d output\_directory] [db\_design\_file2] [-d output\_directory2] ... [db\_design\_fileN] [-d output\_directoryN]  
 generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- ```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI_BPC : [status = not complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSpace] WBI_BSPACE : [status = not complete]
- (6)[SibME] WBI_BPC_ME : [status = not complete]
- (7)[SibME] WBI_CEI_ME : [status = not complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = not complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
- (2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (4)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (5)[save and exit]

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.nd is complete with 0 remaining item(s):
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm_performancedw
- (4)bpm_processserver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi_commonodb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=50000] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating database design files and database scripts (AIX network deployment):

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files and database scripts for deployment environments with DB2 for z/OS:

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and

press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, `WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB`. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
```

Database schema name (SQLID) [default=] :
Bufferpool of 4K size for tables [default=BP1] :
Bufferpool for indexes [default=BP2] :
Bufferpool of 4K size for LOB data [default=BP3] :
Bufferpool of 8K size for tables [default=BP8K1] :
Bufferpool of 16K size for tables [default=BP16K1] :
Bufferpool of 32K size for tables [default=BP32K1] :
Storage group name [default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS  
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database Location on z/OS system [default=] :  
Database server host [default=] :  
Database server port [default=] :  
Data source user name [default=] :  
Data source password [default=] :  
DB2 Universal JDBC driver path [default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
Universal JDBC driver path [default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
DB2 Universal JDBC driver native path (This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4) [default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.
```

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

Note: For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1)false
- (2>true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.nd.topology**, after you configure the master component, you must manually configure the **BPM_PerformanceDW** and **BPM_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :12

[status] bpm.advanced.nd.topology is not complete with 6 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'schemaName' for SCHEMA is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'catalog' for VCAT is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[4] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
[5] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property 'schemaName' for SCHEMA is empty.
[6] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property 'catalog' for VCAT is empty.

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment:

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```



```
-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :

(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- 8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```
- 9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```
- 10. To generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Related tasks:

“Creating the messaging engine data stores” on page 1648

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the

messaging engine data stores.

Generating scripts for creating database objects in DB2 for z/OS:

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
 - Database server details
 - Data source details
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and

generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: cd /opt/IBM/WebSphere/AppServer/util/dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db_design_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

Tip: In the database design file, the directories for the SQL scripts are defined by the *ddl_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

Note: The `createDB2.sh` script requires execute permissions.

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required `userName` and `password` properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Setting up databases for Oracle:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a `BPMDBA` user name for the Process Server database, a `PDWDBA` user name for the Performance Data Warehouse database, and a `CMNDBA` user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```


4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;  
SQL> GRANT execute on DBMS_LOCK to PDWDBA;  
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating database design files for Oracle:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with Oracle:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help  
    display help info.
```

```
-e db_design_file_name  
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```

```
-v db_design_file | db_scripts_output_directory  
    when a db_design_file is given, validation will be done on the specified  
    database design file based on the database specs.
```

When a db_scripts_output_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...  
[db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode.

The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment  
(2)Create a database design for a single component  
(3)Edit an existing database design  
(4)Generate database scripts from a database design  
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology  
(2)bpm.advanced.standalone
```


- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI_BPC : [status = not complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = not complete]
- (6)[SibME] WBI_BPC_ME : [status = not complete]
- (7)[SibME] WBI_CEI_ME : [status = not complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = not complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict.

If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components,  
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.nd.topology**, you must manually configure the **BPM_PerformanceDW**, **BPM_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

```
Please enter the number for the database component :3
```

```
-----  
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named dbDesignGenerator.log is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
```

When a `db_scripts_output_directory` is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...  
[db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment  
(2)Create a database design for a single component  
(3)Edit an existing database design  
(4)Generate database scripts from a database design  
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc  
(2)bpcreporting  
(3)bpm_performancedw  
(4)bpm_processserver  
(5)bspace  
(6)cei  
(7)sca  
(8)sibme  
(9>wbi_commondw
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=CMNDB] :

Database User name[default=] :

Database schema [default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

Database Location(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating databases and database design files for SQL Server:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

Tip: If you are using the federated repositories as a user registry, you can ignore warnings in the `systemout.log` file about maximum key length: ... **Warning! The maximum key length is 900 bytes ...** If you are using the stand-alone LDAP registry, ensure that the number of characters in all the user distinguished name (DN) entries in your organization do not exceed the 131 character limit. If the number of characters in any of the user DN entries exceeds 131 characters, you must change the user account registry to the federated repositories option.

Creating SQL Server databases:

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password  
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name  
'message.routingCache'  
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is  
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class  
[com.lombardiisofware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;  
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;  
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is  
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.  
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement  
(SQLServerPreparedStatement.java:388)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute  
(SQLServerPreparedStatement.java:338)  
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)  
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery  
(SQLServerPreparedStatement.java:281)  
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)  
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)  
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)  
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.  
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmntExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.

- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 124.

| Database management system | Minimum privilege required to use the data store tables | Additional privilege required to create the data store tables |
|----------------------------|--|---|
| Microsoft SQL Server | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege. |

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.

- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating database design files for SQL Server:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with SQL Server:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/uti/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/uti/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
```

help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1) bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.

[info] Completed database components can be edited to change existing or defaulted property values.

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI_BPC : [status = not complete]
- (3) [BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4) [BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI_BSPACE : [status = not complete]
- (6) [SibME] WBI_BPC_ME : [status = not complete]
- (7) [SibME] WBI_CEI_ME : [status = not complete]
- (8) [SibME] WBI_SCA_APP_ME : [status = not complete]
- (9) [SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10) [SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11) [SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI_CommonDB is not complete with 1 remaining item(s):

[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI_CommonDB is complete with 0 remaining item(s):

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the **BPM_ProcessServer**, you must manually configure the **BPM_PerformanceDW** component for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
- (2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (5)[save and exit]

Please enter the number for the database component :2

```
-----  
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with SQL Server in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commond
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :

Data source password[default=] :

MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB  
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Configuring the network deployment environment using the `configureNode` command:

To configure a typical network deployment environment, you can use the `configureNode` command instead of using the Profile Management Tool and Deployment Environment wizard or the `manageprofiles` and `wsadmin` commands. This command provides limited customization by using a properties file, and creates a ready-to-use environment. You can also use `configureNode` later to extend the deployment environment.

Restriction: This command is not supported if your database type is DB2 for z/OS.

Important: The `configureNode` command cannot be run twice on one machine without first deleting the profile that was created. To run the `configureNode` command a second time, first delete the profile.

Creating the deployment environment using the `configureNode` command:

You can use the `configureNode` command to create a typical network deployment environment. The goal of this command is to provide a ready-to-use environment. The only parameters that can be customized are provided in a properties file.

Restriction: This command is not supported if your database type is DB2 for z/OS.

You must have installed the product and created all the databases that are specified in the properties file.

Important: You must have already installed the database software and created the databases (but do not create tables). You must also have created all the users that you specify in the properties file. Make sure the users have the installation privileges listed in Database privileges.

For your SQL Server database server:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
- If connections to the database will be made by the current Windows user that the server is running under, the SQL Server must have **Windows authentication mode** or **SQL Server and Windows Authentication mode** enabled, as specified through Microsoft SQL Server Management Studio.

Restriction: Run `configureNode` commands sequentially; do not try to run the command on multiple machines at the same time.

The `configureNode` command uses default ports to create the deployment manager. You cannot customize the ports using this command. The parameters listed in the sample files are the only parameters that can be customized. Make sure that no other profiles that are configured with default ports are started on the same machine.

The `configureNode` command creates the following network deployment topology:

- A single cell environment
- The environment contains the number of cluster members that you specify in the properties file
- The environment uses the Remote Messaging, Remote Support, and Web topology pattern, which includes the following four clusters:
 - Messaging infrastructure cluster

- Supporting infrastructure cluster
- Application deployment target cluster
- Web application cluster

The command performs the following tasks:

- Creates the deployment manager node based on the values in the deployment manager properties file and starts the deployment manager.
- Creates the deployment environment definition.
- Creates a custom node based on the values in the custom node properties file.
- Federates the node and adds the node to the deployment environment.
- Generates the deployment environment.
- Creates the database tables. (You must already have created the databases. See "Before you begin" for instructions.)
- Runs the bootstrap utility to load the Process Server database with system information.
- Configures Business Space and Process Portal on the web application cluster.

The default port is used to create the deployment manager if the optional ports are not specified in the properties file. If a deployment manager is already running on the machine when you run **configureNode**, creating the deployment environment fails. If a deployment manager has already been created but is not running, you must change the port of one of the deployment managers or you might get a port conflict error when you start the other deployment manager.

To create the deployment environment for the first time, complete the following steps:

1. On the machine where you want to create the deployment environment, locate the sample properties file: *install_root/util/ndUtils/samples/*.
2. Copy the sample files and modify the files to reflect your environment. For example, for a deployment manager for IBM BPM Advanced for Process Server, choose the *sample_adv_pc_dmgr.properties* file.

For each database that you want to create, copy the database parameters section and specify the database name, user name, password, and schema name. The Process Server database, Performance Data Warehouse database, and Common database are required, and they must have different names. For IBM BPM Advanced, the Business Process Choreographer database is also required.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

Restriction: The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the names of the two files you created. The custom node file is optional, but if you do not include it, only the deployment manager is created. For example:

```
install_root/util/ndUtils/configureNode.sh -dmgr_response my_dmgr_response_file.properties
-response node01_response_file.properties
```

Note: If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to **127.0.0.1 localhost** in the hosts file. For example: **127.0.0.1 localhost example.ibm.com**.

You now have a deployment manager, a custom node, and a deployment environment that use the Remote Messaging, Remote Support, and Web topology pattern.

Messages are recorded in the file *install_root/logs/config/configureNode.log*.

Tip: After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception

`com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT.` This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

 Virtual hosts

Related tasks:

 Configuring virtual hosts

Related reference:

 Test connection service

Extending the deployment environment using the `configureNode` command:

After you have created a deployment environment using the **configureNode** command, you can run the command at a later time to extend the deployment environment.

You need the following prerequisites before running the **configureNode** command:

- You must have installed the product on the machine where you want to extend the deployment environment.
- The deployment manager must be running.
- You must have created the deployment environment by running the **configureNode** command.

Restriction: Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.

1. On the machine where you want to extend the deployment environment, locate the sample custom node files: `install_root/util/ndUtils/samples/`
2. Copy the sample file that is most similar to the properties file that you need. For example, to extend the deployment environment for IBM BPM Advanced for Process Server, choose the `sample_adv_ps_node.properties` file.

Modify the new custom node file to reflect your environment, including the details of the deployment manager.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

Restriction: The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the name of the sample file. For example:

`install_root/util/ndUtils/configureNode.sh -response node01_response_file.properties`

Note: If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to `127.0.0.1 localhost` in the hosts file. For example: `127.0.0.1 localhost example.ibm.com`.

You have extended the deployment environment.

Messages are recorded in the file `install_root/logs/config/configureNode.log`.

Tip: After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception `com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT`. This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

 [Virtual hosts](#)

Related tasks:

 [Configuring virtual hosts](#)

Related reference:

 [Test connection service](#)

Configuring profiles and creating a network deployment environment:

After you install the product, you must create or augment a deployment manager and one or more custom profiles to define the runtime environment. Before starting the deployment manager, you must have configured the databases that are to be used with IBM Business Process Manager.

Creating or augmenting network deployment profiles:

If you used the **configureNode** command to create and configure your deployment environment, no further configuration is required. Otherwise, you must create or augment a deployment manager profile and one or more custom profiles before creating the deployment environment. Using profiles, you can have more than one runtime environment on a system, without having to install multiple copies of IBM Business Process Manager.

Creating or augmenting network deployment profiles with a DB2 database server:

You can configure a network deployment environment for IBM Business Process Manager using a DB2 database server.

Preparing the DB2 database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components. You can either create a new database and tables or postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

If you plan to configure the Common database on a remote server, you must have created the database before beginning to create or augment the profile. You can create a database on the local server or use an existing one on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including `PKCS12`, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

profile_root/properties/portdef.prop

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.
 - e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 125. Required Common database configuration fields for DB2 Database

| Field | Action needed |
|---|--|
| User name to authenticate with the database | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> . |
| Database server host name (for example IP address) | Accept the default value of <code>localhost</code> or enter the correct database server host name. |
| Server port | Accept the default value of <code>50000</code> or enter the correct server port number. |
| Schema name | Enter a schema name for the Common database. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with a DB2 database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```


For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

profile_root/properties/portdef.prop

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - e. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 126. Required Common database configuration fields for DB2 Database

| Field | Action needed |
|---|--|
| User name to authenticate with the database | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> . |
| Database server host name (for example IP address) | Accept the default value of <code>localhost</code> or enter the correct database server host name. |
| Server port | Accept the default value of <code>50000</code> or enter the correct server port number. |
| Schema name | Enter a schema name for the Common database. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create a deployment manager profile.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the `-templatePath` option).

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:

- `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.

- dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response

- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
```



```

# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```

```

# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: keyStorePassword
#

```

```

# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service

```

```

# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous

```

```

# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#

```



```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that

```

```

# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#

```

```

# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid

```

```

#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts

```

```

# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#

```

```

# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false

```



```

#winserviceStartupType>manual

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM

```

```

# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a

```

```

# trailing number, for example:
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#    Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#    For IBM Business Process Manager omit this parameter. Administrative
#    security is always enabled for IBM Business Process Manager profiles.
#    You must also specify the parameters -adminUserName and -adminPassword
#    along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#    Specifies the distinguished name of the root signing certificate that you
#    create when you create the profile. Specify the distinguished name in
#    quotation marks. This default personal certificate is located in the
#    server keystore file. If you do not specifically create or import a root
#    signing certificate, one is created by default. See the
#    -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#    An optional parameter that specifies the amount of time in years that the
#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the

```

```

#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#

```

```

# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####

```



```

# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbHostName
#
# Description:
# The database server host name or IP address. The default value is the
# local host name. Required when the -bpmdbDesign parameter was not
# specified. (It is not required if the -bpmdbDesign parameter is
# specified.)
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbServerPort
#
# Description:
# The database server port number. Depending on the database you are using,
# you can specify a different port number instead of the default port
# number.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdriver/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
```

```

# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```

```

# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: keyStorePassword
#

```

```

# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies

```



```

# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service

```

```

# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a

```

```

# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with a DB2 database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password.

The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 127. Required Common database configuration fields for DB2 Database

| Field | Action needed |
|---|--|
| User name to authenticate with the database | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> . |
| Database server host name (for example IP address) | Accept the default value of <code>localhost</code> or enter the correct database server host name. |
| Server port | Accept the default value of <code>50000</code> or enter the correct server port number. |
| Schema name | Enter a schema name for the Common database. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles for Process Server with a DB2 database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. If you selected **Typical** profile creation, skip to the Administrative Security step.
8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
9. **Advanced:** On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. **Required:** On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 128. Required Common database configuration fields for DB2 Database

| Field | Action needed |
|---|--|
| User name to authenticate with the database | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> . |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |

Table 128. Required Common database configuration fields for DB2 Database (continued)

| Field | Action needed |
|-------------|--|
| Server port | Accept the default value of 50000 or enter the correct server port number. |
| Schema name | Enter a schema name for the Common database. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.

- dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response

- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing

certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response

- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
```

```

# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the

```

```

#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  enableService
#
# Description:
#    Enables the creation of a Linux service. Valid values include true or
#    false. The default value for this parameter is false. When the
#    manageprofiles command-line utility is run with the -enableService option
#    set to true , the Linux service is created with the profile when the
#    command is run by the root user. When a nonroot user runs the
#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#    Specifies the user ID that is used during the creation of the Linux
#    service so that the Linux service will run under this user ID. The Linux
#    service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#    The type of the owner account of the Windows service created for the
#    profile. Valid values include specifieduser or localsystem. The
#    localsystem value runs the Windows service under the local account of the

```

```

# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled

```



```

# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.

```

```

# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values

```

```

# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a

```

```

# Managed PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be

```

```

# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#

```

```

# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:

```



```

# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the

```

```

# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.

```

```

# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#

```

```

# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#

```



```

# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER

```

```
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
```

```
#####  
# Parameter: federateLaterBPM  
#  
# Description:  
# Indicates if the managed profile is to be federated later using either the  
# profileTemplates/managed.esbserver or the  
# profileTemplates/BPM/managed.templateName template. Valid values are true  
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and  
# -dmgrAdminPassword parameters are not set, the default value for this  
# parameter is true. If you use -federateLaterBPM, you must either omit the  
# WebSphere Application Server-based parameter -federateLater or set it to  
# true.  
#  
# Valid Values:  
# false  
# true  
# Default Values:  
# true  
#  
# Parameter: dmgrHost  
#  
# Description:  
# Identifies the workstation where the deployment manager is running.  
# Specify this parameter and the dmgrPort parameter to federate a custom  
# profile as it is created or augmented. This parameter is available with  
# the managed.templateName and managed.esbserver profile templates. The  
# host name can be the long or short DNS name or the IP address of the  
# deployment manager workstation. Specifying this optional parameter directs  
# the manageprofiles command-line utility to attempt to federate the custom  
# node into the deployment manager cell as it creates the custom profile.  
# This parameter is ignored when creating a deployment manager profile or  
# stand-alone server profile. If you federate a custom node when the  
# deployment manager is not running, the installation indicator in the logs  
# is INSTCONFFAILED to indicate a complete failure. The resulting custom  
# profile is unusable. You must move the custom profile directory out of the  
# profile repository (the profile's installation root directory) before  
# creating another custom profile with the same profile name. If you have  
# changed the default JMX connector type, you cannot federate with the  
# manageprofiles command-line utility. Use the addNode command later after  
# the node is created instead. The default value for this parameter is  
# localhost. The value for this parameter must be a properly formed host  
# name and must not contain spaces or characters that are not valid such as  
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the  
# deployment manager must also be available in conjunction with the dmgrPort  
# parameter. Required if you are creating a managed profile with the  
# -federateLaterBPM parameter set to false and the deployment manager is not  
# on the local machine.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#  
# Parameter: dmgrPort  
#  
# Description:  
# Identifies the SOAP port of the deployment manager. Specify this parameter  
# and the dmgrHost parameter to federate a custom profile as it is created  
# or augmented. The deployment manager must be running and accessible. If  
# you have changed the default JMX connector type, you cannot federate with  
# the manageprofiles command-line utility. Use the addNode command later  
# after the node is created instead. The default value for this parameter is  
# 8879. The port that you indicate must be a positive integer and a  
# connection to the deployment manager must be available in conjunction with  
# the dmgrHost parameter. Required if you are creating a managed profile,
```

```

# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```
#####  
templatePath=BPM/managed.procsvr
```

```
#####
```

```
# Parameter: profileName
```

```
#
```

```
# Description:
```

```
# Specifies the name of the profile. Use a unique value when creating a  
# profile.  
# Each profile that shares the same set of product binaries must have a  
# unique name. The default profile name is based on the profile type and a  
# trailing number, for example:  
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,  
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a  
# unique profile name. The value for this parameter must not contain spaces  
# or characters that are not valid such as the following: *, ?, ", <, >, ,,  
# /, \, and |. The profile name that you choose must not be in use. On  
# Windows platforms: If the fully qualified path contains spaces, enclose  
# the value in quotation marks. The default value is based on the  
# install_root directory, the profiles subdirectory, and the name of the  
# file. For example, the default for profile creation is:  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties  
# file in the install_root/properties directory. The value for this  
# parameter must be a valid path for the target system and must not be  
# currently in use. You must have permissions to write to the directory.  
#
```

```
# Valid Values:
```

```
# Varies
```

```
# Default Values:
```

```
# None
```

```
#####
```

```
profileName=Custom01
```

```
#####
```

```
# Parameter: cellName
```

```
#
```

```
# Description:
```

```
# Specifies the cell name of the profile. Use a unique cell name for each  
# profile.  
# The default value for this parameter is based on a combination of the  
# short host name, the constant Cell, and a trailing number, for example:  
# if (Dmgr)  
#     shortHostNameCellCellNumber  
# else  
#     shortHostNameNodeNodeNumberCell  
# where CellNumber is a sequential number starting at 01 and NodeNumber is  
# the node number that you used to define the node name.  
# The value for this parameter must not contain spaces or any characters  
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.  
#
```

```
# Valid Values:
```

```
# Varies
```

```
# Default Values:
```

```
# Varies
```

```
#
```

```
# Parameter: nodeName
```

```
#
```

```
# Description:
```

```
# Specifies the node name for the node that is created with the new profile.  
# Use a unique value within the cell or on the workstation. Each profile  
# that shares the same set of product binaries must have a unique node name.  
# This parameter is required for profile creation only with the  
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.  
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.  
# The default value for this parameter is based on the short host name,  
# profile type, and a trailing number, for example:
```

```

#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile

```

```

# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you

```



```

# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: federateLaterBPM

```

```

#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command **install_root/bin/ProfileManagement/pmt.sh**.
 3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:
 - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
 - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.

- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response

- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:


```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):


```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

Creating or augmenting network deployment profiles with a DB2 for z/OS database server:

You can configure a network deployment environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.

- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
14. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
15. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool on AIX:

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
14. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
15. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create new profiles.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the `-templatePath` option).

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:

 - `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.
 - `dmgr.procctr.adv`: for a Process Center deployment manager profile for IBM BPM Advanced.
 - `dmgr.procsvr`: for a Process Server deployment manager profile for IBM BPM Standard.

- dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response

- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWdbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
```

```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber

```

```

#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

```



```

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#    Indicates a new database is created or if an existing database is to be
#    reused. Valid values are true or false. This value cannot be true for an
#    Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#    a DB2 database.
#
# Valid Values:
#    false
# Default Values:
#    false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#    The database type. Set one of the following values for the type of
#    database product you are using with IBM Business Process Manager.
#
#    Required only for one of the following:
#    A stand-alone profile when -bpmdbDesign was not specified.
#    An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2UDBOS390
# Default Values:
#    DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#    Indicates whether to postpone table creation until after the profile is
#    created. Valid values are true or false. The parameter is set to false by
#    default. Setting this parameter to true delays the execution of the
#    scripts that create the database tables. No. Specify false if you want the
#    database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#    The name of the database. By default, the value is set to orcl for Oracle
#    databases and to CMNDB for all other supported databases.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

```

```

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94

```

```

# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:

```

```

# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies

```



```

#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option

```

```

# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:

```

```

# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.

```

```

#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on

```

```

# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

```

```

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

```



```

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.

```

```

# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux

```

```

#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter: winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#

```

```

# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:

```

```

# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2UDBOS390
# Default Values:
# DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####

```

```
environmentType=Test
```

```
#####  
# Parameter: isDeveloperServer  
#  
# Description:  
#   Specifies whether the server is intended for development purposes only.  
#   This parameter is useful when creating profiles to test applications on a  
#   non-production server prior to deploying the applications on their  
#   production application servers. If -isDeveloperServer is set when  
#   creating an IBM Business Process Manager profile, then a preconfigured VMM  
#   file repository is installed. This file repository contains a sample  
#   organization that can be used to test Business Process Choreographer  
#   people resolution, ready for you to use as is.  
#  
# Valid Values:  
#   false  
#   true  
# Default Values:  
#   false  
#####  
#isDeveloperServer=false
```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT  
# *****  
#  
# Licensed Materials - Property of IBM  
# 5725-C94  
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.  
# This sample program is provided AS IS and may be used, executed, copied  
# and modified without royalty payment by customer (a) for its own  
# instruction and study, (b) in order to develop applications designed to  
# run with an IBM WebSphere product, either for customer's own internal  
# use or for redistribution by customer, as part of such an application,  
# in customer's own products.  
#  
# *****  
# END COPYRIGHT  
  
#####  
# These response file has the applicable parameters for creating a  
# DMgr PS Std profile that uses DB2zOS.  
#  
# Depending on your environment, you may need to change the default values.  
#  
# To create a profile with this response file specify:  
#  
# was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2zOS.response  
#  
# If you use the -response parameter, it must be the only parameter on  
# the command-line. If you include any manageprofile parameters  
# (in addition to -response) on the command-line, the response file  
# is ignored. And default WebSphere Application server profile will be created .  
#####  
  
#####  
# Parameter: create  
#
```

```

# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:

```



```

#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root

```

```

# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType

```

```

#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.

```

```

#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command ***install_root*/bin/ProfileManagement/pmt.sh**.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 8. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

14. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.

17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command **install_root/bin/ProfileManagement/pmt.sh**.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
8. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page
 When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

14. Advanced: Configure the Common database using a design file.

- a. Select **Use a database design file**.
- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.

- d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response

- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```


After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation If you choose to federate the

node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The

password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the `manageprofiles` command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
 - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
 - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
 - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
 - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`

- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
```

```

# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the

```

```

#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the

```

```

# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled

```



```

# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter:  federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false.  If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true.  If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter:  dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before

```

```

# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test

```

```

# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

```

```

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:

```

```

# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:

```

```

# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#

```

```

# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true

```

```

# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is

```



```

# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, ,, /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test

```

```
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
```

```

# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:

```

```

#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root

```

```

# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType

```

```

#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390
#####
# Parameter:  federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false.  If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true.  If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter:  dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom

```

```

# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:

```



```

# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT
#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2zOS.

```

```

#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:

```

```

# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log

```

```

# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs

```

```

# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage

```

```

# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.

2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
3. Shut down any servers associated with a profile you plan to augment.
4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.

16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 Updating ports in existing profiles (WebSphere Application Server)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

3. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
4. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

5. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response

- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

6. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:

```
profile_root/bin/startManager.sh
```
- The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
- The deployment manager is at the same release level or higher than the custom profile you created or augmented.

- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

Creating and configuring DB2 for z/OS databases after network deployment profile creation:

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

DB2 for z/OS authorization prerequisites:

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as *#DB_USER#*. These permissions are available in the *createDatabase.sql* database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;  
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;  
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the *#DB_USER#* user to create sequences and stored procedures with a schema qualifier of *#SCHEMA#*:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;  
GRANT BINDADD TO #DB_USER#;
```

Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database #DB_NAME# can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.EVENT TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;
```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named SICELL, and connect user WSADMIN to it, as follows:


```
INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS  OWNER=ZWPS  CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
  WSADMIN  CONNECT  000000  NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE  RESUME DATE=NONE
```

Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832
Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating databases in the DB2 for z/OS subsystem:

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

Considerations for choosing the createDB2.sh script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two .sql files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

Related tasks:

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Configuring the DB2 command line processor:

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
 - Modify the CLASSPATH environment variable to include the `clp.jar` file.
 - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
 - Define the **db2** command as an alias for the command that starts the command line processor.
 - Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:

- A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
- A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

Tip: When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script”

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 1644

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z_schema.sql and z_schemaProc.sql) that you can run later by using the DB2 command line processor.

Creating DB2 for z/OS database objects using the createDB2.sh script:

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTPE2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database

configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCreate  
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

-RunSQL

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

Tip: The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 1641

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using the DB2 command line processor:

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database

configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate  
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCcreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

Tip: By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two .sql files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP  
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

The z_schema.sql and z_schemaProc.sql files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The z_schemaProc.sql file contains SQL statements for stored procedures, and is generated in addition to z_schema.sql because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the z_schema.sql file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the z_schemaProc.sql file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 1641

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

Tip: The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

Tip: If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

```
@STOGRP@ = DB2 Storage group name
@DB_NAME@ = DB2 Database name
@SCHEMA@ = DB2 SQLID (Schema Qualifier)
@BP4K@ = Buffer Pool of 4k Size
@BPINDEX@ = Buffer Pool Index
@VOLUMES@ = DB2 comma separated list of volume-ids
@VCAT@ = DB2 integrated catalog facility name
@DB_USER@ = The user ID which will be used to create the database objects
```

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

Tip: If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (z_schema.sql and z_schemaProc.sql), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

Related tasks:

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 1644

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z_schema.sql and z_schemaProc.sql) that you can run later by using the DB2

command line processor.

Creating the messaging engine data stores:

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Before you create and run the database scripts:

- Create a cluster.
- Decide which service integration bus components you want to create in your network deployment configuration.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
 - When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
 - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
 - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
 - a. Run the database design tool (**DbDesignGenerator**) from the *install_root/utl/dbUtils* directory.
 - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
 - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

Tip: As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

Related tasks:

“Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment” on page 1420

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for deployment environments with DB2 for z/OS” on page 1413

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

Note: Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

Setting the correct schema name for the messaging engines:

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
 - a. Select **Messaging engines**, and then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Verify that the messaging engines can be started:

- a. Log out of the administrative console.
- b. Stop, and then restart the server.
- c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

Creating or augmenting network deployment profiles with an Oracle database server:

You can configure a network deployment environment for IBM Business Process Manager using an Oracle database server.

Preparing the Oracle database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

To configure the Common database while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default

language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.

- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

d. Click **Next**.

15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 129. Required database configuration fields for Oracle - Part 2

| Field | Action needed |
|---|--|
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 130. Required database configuration fields for Oracle - Part 2

| Field | Action needed |
|---|---|
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
- dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
- dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
- dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
- dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.

2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response

- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.


```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
# Parameter: profileName
#
# Description:

```

```

# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumber where profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService

```

```

#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

```



```

#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword

```

```

#
# Description:
#   The password required for database authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test

```

```

# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .

```

```

#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#

```

```

# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#

```

```

# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port

```

```

# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:

```



```

#    Varies
# Default Values:
#    None
#
# Parameter: winserviceAccountType
#
# Description:
#    The type of the owner account of the Windows service created for the
#    profile. Valid values include specifieduser or localsystem. The
#    localsystem value runs the Windows service under the local account of the
#    user who creates the profile. The default value for this parameter is
#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:

```

```

# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbDriverType=ORACLE_THIN

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test

```

```
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
```

```

#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the

```

```

# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the

```

```

# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the

```

```

# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```



```

# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled

```

```

# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE

```

```

# Default Values:
#   ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter:  dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location
#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####

```

dbDelayConfig=false

```
#####  
# Parameter: dbName  
#  
# Description:  
#   The name of the database. By default, the value is set to orcl for Oracle  
#   databases and to CMNDB for all other supported databases.  
#  
#   Note: Do not use this parameter if you are using -bpmdbDesign.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   CMNDB  
#  
# Parameter: dbUserId  
#  
# Description:  
#   The user ID for all database types. Specifies the user ID that has  
#   privileges to create and drop the databases. The WebSphere data source  
#   uses this ID to authenticate the database connection.  
#   Important: The -dbUserId parameter value must be a valid database  
#   authorization ID.  
#  
#   Note: Do not use this parameter if you are using -bpmdbDesign.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: dbPassword  
#  
# Description:  
#   The password required for database authentication.  
#  
#   Note: Do not use this parameter if you are using -bpmdbDesign.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#####  
#dbName=orcl  
#dbUserId=db_userid  
#dbPassword=db_pswd  
#####  
# Parameter: dbCommonUserId  
#  
# Description:  
#   The user id to authenticate with the Common database.  
#  
#   Note: Do not use this parameter if you are using -bpmdbDesign.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: dbCommonPassword  
#  
# Description:  
#   The password for the user id of the dbCommonUserId parameter.  
#  
#   Note: Do not use this parameter if you are using -bpmdbDesign.
```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
```

```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber

```

```

# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:

```



```

#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

```

```

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with an Oracle database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.

9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 131. Required database configuration fields for Oracle - Part 2

| Field | Action needed |
|---|---|
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles for Process Server with an Oracle database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command **install_root/bin/ProfileManagement/pmt.sh**.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. **Advanced:** On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. **Required:** On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 132. Required database configuration fields for Oracle - Part 2

| Field | Action needed |
|---------------------------|--|
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |

Table 132. Required database configuration fields for Oracle - Part 2 (continued)

| Field | Action needed |
|---|---|
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.

- dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response

- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the **bin** directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.

14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response

- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters.
For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
```

create

```
#####  
# Parameter: templatePath  
#  
# Description:  
# Specifies the directory path to the template files in the installation  
# root directory. Within the profileTemplates directory are various  
# directories that correspond to different profile types and that vary with  
# the type of product installed. The profile directories are the paths that  
# you indicate while using the -templatePath option.  
# Use absolute paths. This parameter must exist as a directory and point to  
# a valid template directory.  
#  
# When using the -templatePath parameter, specify the fully qualified file  
# path for the parameter.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
templatePath=BPM/managed.procctr.adv
```

```
#####  
# Parameter: profileName  
#  
# Description:  
# Specifies the name of the profile. Use a unique value when creating a  
# profile.  
# Each profile that shares the same set of product binaries must have a  
# unique name. The default profile name is based on the profile type and a  
# trailing number, for example:  
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,  
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a  
# unique profile name. The value for this parameter must not contain spaces  
# or characters that are not valid such as the following: *, ?, ", <, >, ,,  
# /, \, and |. The profile name that you choose must not be in use. On  
# Windows platforms: If the fully qualified path contains spaces, enclose  
# the value in quotation marks. The default value is based on the  
# install_root directory, the profiles subdirectory, and the name of the  
# file. For example, the default for profile creation is:  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties  
# file in the install_root/properties directory. The value for this  
# parameter must be a valid path for the target system and must not be  
# currently in use. You must have permissions to write to the directory.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
profileName=Custom01
```

```
#####  
# Parameter: cellName  
#  
# Description:  
# Specifies the cell name of the profile. Use a unique cell name for each  
# profile.  
# The default value for this parameter is based on a combination of the  
# short host name, the constant Cell, and a trailing number, for example:  
# if (Dmgr)  
# shortHostNameCellCellNumber  
# else  
# shortHostNameNodeNodeNumberCell
```

```

# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:

```



```

#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual

```

```

#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  dbType
#
# Description:
#    The database type. Set one of the following values for the type of
#    database product you are using with IBM Business Process Manager.
#
#    Required only for one of the following:
#    A stand-alone profile when -bpmdbDesign was not specified.
#    An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    ORACLE
# Default Values:
#    ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#    The database driver type. For an Oracle database, specify ORACLE. For
#    databases other than Oracle, the value is automatically set based on the
#    server operating system. Server installations on all operating systems use
#    type 4.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#    The directory path location of JDBC driver files.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

```

```

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dmgrPort
#
# Description:
# Identifies the SOAP port of the deployment manager. Specify this parameter
# and the dmgrHost parameter to federate a custom profile as it is created
# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment

```

```

# manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```

#####
templatePath=BPM/managed.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:

```



```

#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile

```

```

# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you

```

```

# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbDriverType

```

```

#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: dmgrPort
#
# Description:
# Identifies the SOAP port of the deployment manager. Specify this parameter
# and the dmgrHost parameter to federate a custom profile as it is created
# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when

```

```

# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####
#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
# Specifies the directory path to the template files in the installation

```

```

#    root directory. Within the profileTemplates directory are various
#    directories that correspond to different profile types and that vary with
#    the type of product installed. The profile directories are the paths that
#    you indicate while using the -templatePath option.
#    Use absolute paths. This parameter must exist as a directory and point to
#    a valid template directory.
#
#    When using the -templatePath parameter, specify the fully qualified file
#    path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv
#####

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01
#####

# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:

```



```

# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The

```

```

#    localsystem value runs the Windows service under the local account of the
#    user who creates the profile. The default value for this parameter is
#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic

```

```

# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the

```

```

# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dmgrPort
#
# Description:
# Identifies the SOAP port of the deployment manager. Specify this parameter
# and the dmgrHost parameter to federate a custom profile as it is created
# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
# Varies
# Default Values:
# None
#####

```

```

federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter: profileName
#
# Description:

```

```

# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumber where profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```



```

#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not

```

```

# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created

```

```

# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false

```

```
# true
# Default Values:
# false
#####
#isDeveloperServer=false
```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.

- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.

- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

- Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

- Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

Augmenting custom profiles using the `manageprofiles` command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the `manageprofiles` command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - *managed.procctr*: for a Process Center custom profile for IBM BPM Standard.
 - *managed.procctr.adv*: for a Process Center custom profile for IBM BPM Advanced.
 - *managed.procsvr*: for a Process Server custom profile for IBM BPM Standard.
 - *managed.procsvr.adv*: for a Process Server custom profile for IBM BPM Advanced.
 - *managed.esbserver*: for a WebSphere Enterprise Service Bus custom profile.
3. Use the *augment* parameter to make changes to an existing profile with an augmentation template. The *augment* parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- *PC_Std_DMGr_DB2.response*
- *PC_Std_DMGr_DB2z0S.response*
- *PC_Std_DMGr_Oracle.response*
- *PC_Std_DMGr_SQLServer.response*
- *PC_Adv_DMGr_DB2.response*
- *PC_Adv_DMGr_DB2z0S.response*
- *PC_Adv_DMGr_Oracle.response*
- *PC_Adv_DMGr_SQLServer.response*
- *PC_Std_Managed_DB2.response*
- *PC_Std_Managed_DB2z0S.response*
- *PC_Std_Managed_Oracle.response*
- *PC_Std_Managed_SQLServer.response*
- *PC_Adv_Managed_DB2.response*
- *PC_Adv_Managed_DB2z0S.response*
- *PC_Adv_Managed_Oracle.response*
- *PC_Adv_Managed_SQLServer.response*
- *PC_Std_StandAlone_DB2.response*
- *PC_Std_StandAlone_DB2z0S.response*

- PC_Std_Standalone_Oracle.response
- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
 -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:

```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 Add managed node settings (WebSphere Application Server)

Creating or augmenting network deployment profiles with an SQL Server database server:

You can configure a network deployment environment for IBM Business Process Manager using a Microsoft SQL Server database server.

Preparing the SQL Server database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Restriction: The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 133.

| Database management system | Minimum privilege required to use the data store tables | Additional privilege required to create the data store tables |
|----------------------------|--|---|
| Microsoft SQL Server | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege. |

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.

- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
 - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
 - c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles with an SQL Server database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.

- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 134. Required database configuration fields for Microsoft SQL Server - Part 2

| Field | Action needed |
|---|--|
| Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. | |
| Common database user name | Enter the user name to authenticate with the database. |

Table 134. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

| Field | Action needed |
|---|---|
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with an SQL Server database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).

- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all

internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 135. Required database configuration fields for Microsoft SQL Server - Part 2

| Field | Action needed |
|---|--|
| Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. | |
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |

Table 135. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

| Field | Action needed |
|--|---|
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_Standalone_DB2.response
- PC_Std_Standalone_DB2zOS.response
- PC_Std_Standalone_Oracle.response
- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response

- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUser perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a deployment manager profile using the **manageprofiles** command that includes the following components: CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space.

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates\BPM\
\dmgr.procsvr.adv -dbHostName hostName -dbServerPort 1433 -dbDelayConfig true
-configureBspace true -dbType MSSQLSERVER_Microsoft -dbUserId userID -dbJBCClasspath
path to JDBC 2.0 driver -dbName CMNDB
-dbPassword password
```

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
```

```

# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a

```



```

# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period

```

```

#keyStorePassword =keyStore_password

#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or

```

```

# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword

```

```

#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

```

```

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

```

```

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
#dbUserId=db_userid
#dbPassword=db_pswd
#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```



```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample

```

```

# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
# Specifies the directory path to the template files in the installation
# root directory. Within the profileTemplates directory are various

```

```

# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the

```

```

#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port

```

```

# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:

```

```

#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID

```

```

#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:

```



```

# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to

```

```

# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv
#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01
#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#

```

```

# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an

```

```

#      automatically generated set of recommended ports if you do not specify the
#      -startingPort parameter, the -defaultPorts parameter or the -portsFile
#      parameter. The recommended port values can be different than the default
#      port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#

```

```

# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```

```

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

```

```

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####

```



```

# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#

```

```

# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters

```

```

# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####

```

```

# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####

```

```

# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,

```

```

#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType

```

```

#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage

```



```

# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with an SQL Server database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12`

file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 136. Required database configuration fields for Microsoft SQL Server - Part 2

| Field | Action needed |
|--|---|
| | Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. |
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles for Process Server with an SQL Server database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```


For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root*/bin/ProfileManagement/pmt.sh.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*/profiles/*profile_name*.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration _ Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.
- If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 137. Required database configuration fields for Microsoft SQL Server - Part 2

| Field | Action needed |
|---|--|
| Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive. | |
| Common database user name | Enter the user name to authenticate with the database. |
| Common database password | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Related tasks:

Loading the database with system information in a network deployment environment

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response

- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command ***install_root*/bin/ProfileManagement/pmt.sh**.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones.

If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:
 - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
 - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
 - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
 - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
 - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response

- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
```

```

#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the

```

```

# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#

```

```

# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option

```

```

# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:

```

```

# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:

```

```

# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true
#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have

```

```

#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production

```



```

# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

```

```

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each

```

```

# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you

```

```

# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:

```

```

# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:

```

```

# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbWinAuth
#
# Description:
# An optional parameter that is valid in all default profile templates.
# Specify true to indicate that you are running SQL Server using integrated
# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true

# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.

```

```

# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values

```

```

# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a

```



```

# Managed PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be

```

```

# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#

```

```

# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# MSSQLSERVER_MICROSOFT
# Default Values:
# MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true
#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the

```

```

# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

```

```
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false
```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
```



```

#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else

```

```

#       shortHostNameNodeNodeNumberCell
#       where CellNumber is a sequential number starting at 01 and NodeNumber is
#       the node number that you used to define the node name.
#       The value for this parameter must not contain spaces or any characters
#       that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:

```

```

# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbWinAuth
#
# Description:
# An optional parameter that is valid in all default profile templates.
# Specify true to indicate that you are running SQL Server using integrated
# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom

```

```

# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a

```

```

#    preproduction server.
#    Production
#    Use Production if the server is to serve in a production capacity.
#    The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#    Specifies whether the server is intended for development purposes only.
#    This parameter is useful when creating profiles to test applications on a
#    non-production server prior to deploying the applications on their
#    production application servers. If -isDeveloperServer is set when
#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.

2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
3. Shut down any servers associated with a profile you plan to augment.
4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

8. If you selected **Typical** profile creation, skip to the Federation step.
9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.

16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 Updating ports in existing profiles (WebSphere Application Server)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2z0S.response

- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:


```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):


```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```

An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:



Add managed node settings (WebSphere Application Server)

Loading the database with system information in a network deployment environment:

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly

- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. For a cluster, you need to specify the cluster name. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.
- If a single WebSphere cell contains multiple application target clusters, you must run this command on each of the clusters.

Run the bootstrap utility from the command line. The bootstrap utility is found in the deployment manager profile directory. For example:

```
Linux UNIX install_root/profiles/Dmgr01/bin
```

```
Windows install_root\profiles\Dmgr01\bin
```

Run the bootstrap utility using one of the following commands:

- **bootstrapProcessServerData.sh -clusterName** *cluster_name*
- **bootstrapProcessServerData.sh -nodeName** *node_name* **-serverName** *server_name*

where:

- **-clusterName** is the name of the application target cluster. You must specify this parameter when you want the bootstrap data to run on a cluster.
- **-nodeName** is the name of the node. You must specify this parameter and the **-serverName** parameter when you want the bootstrap data to run on the server that is part of the network deployment environment and not part of the cluster.

- **-serverName** is the name of the server. You must specify either this parameter and the **-nodeName** parameter when you want the bootstrap data to run on the server that is part of the network deployment and not part of the cluster.

The parameters are case-sensitive.

Important: For SQL Server only: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

You have loaded the database with system information before successfully starting the IBM Business Process Manager server. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.clusterName.timestamp.log` or `bootstrapProcessServerData.nodeName.serverName.timestamp.log` depending on the target you specified. The console displays a subset of the logged information.

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData
```

Bootstrap data onto a cluster that hosts the Process Server or Process Center:

```
bootstrapProcessServerData -clusterName myAppCluster
```

Modifying an existing installation:

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

Installing and configuring IBM Business Process Manager on Solaris

Install IBM Business Process Manager on Solaris, and configure a stand-alone or network deployment environment.

Configuring a stand-alone environment on Solaris

You can use a typical installation to install the software and configure a stand-alone Process Center or Process Server profile. Or you can choose custom installation and then create profiles, giving you more control over how to configure your software.

Typical installation and configuration:

The Typical installation option is the simplest and quickest method for installing and configuring IBM Business Process Manager.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing and configuring IBM Business Process Manager with a DB2 database:

You can install IBM Business Process Manager using a DB2 database server. You must make sure the required databases are available before you install.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

Installing and configuring Process Center with a DB2 database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 138. Required database configuration fields for DB2

| Field | Action needed |
|--|--|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 50000 or enter the correct server port number. |
| Username | Enter the user name to authenticate with the database.
Restriction: User names must not contain National Language Strings (NLS). |
| Password | Enter a password to authenticate with the database. |
| Common database name | Enter the name for the Common database. |
| Process Center database name | Enter the name for the Process Center database. |
| Performance Data Warehouse database name | Enter the name for the Performance Data Warehouse database. |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with a DB2 database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.

- Select **Test** if the server is to be used as a testing environment, for example, for load testing.
- **Environment Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 139. Required database configuration fields for DB2

| Field | Action needed |
|---|--|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 50000 or enter the correct server port number. |
| Username | Enter the user name to authenticate with the database.
Restriction: User names must not contain National Language Strings (NLS). |
| Password | Enter a password to authenticate with the database. |
| Common database name | Enter the name for the Common database. |
| Process Server database name | Enter the name for the Process Server database. |
| Performance Data Warehouse database name | Enter the name for the Performance Data Warehouse database. |

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license**

agreement and notices. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with an Oracle database server:

You can install IBM Business Process Manager using an Oracle database server. You must make sure the required databases are available before you install.

Creating Oracle databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

Installing and configuring Process Center with an Oracle database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication

- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.


```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of `localhost` or `127.0.0.1` is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 140. Required database configuration fields for Oracle

| Field | Action needed |
|--|--|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 1521 or enter the correct server port number. |
| Database name | Enter the correct Process Center database System Identifier. |
| Common database | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database. |
| Process Center database | For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database. |
| Performance Data Warehouse database | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database. |
| Common Event Infrastructure bus messaging engine database | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine. |
| Service Component Architecture system bus messaging engine database | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine. |
| Service Component Architecture application bus messaging engine database | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine. |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with an Oracle database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```


Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses three prefixes: `ifix`, `fixpack`, and `launchpad`. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.
`extract_directory/launchpad.sh`
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - **Environment Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 141. Required database configuration fields for Oracle

| Field | Action needed |
|---|--|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 1521 or enter the correct server port number. |
| Database name | Enter the correct Process Center database System Identifier. |
| Common database | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database. |
| Process Server database | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database. |
| Performance Data Warehouse database | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database. |
| Common Event Infrastructure bus messaging engine database | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine. |
| Service Component Architecture system bus messaging engine database | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine. |
| Service Component Architecture application bus messaging engine database | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine. |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with an SQL Server database server:

You can install IBM Business Process Manager using a Microsoft SQL Server database server. You must make sure the required databases are available before you install.

Creating SQL Server databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password  
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database,

and `database_name` is the name of the database that you are creating. `COLLATE SQL_Latin1_General_CP1_CI_AS` is included because these databases must be case-insensitive.

b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the `COLLATE` clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters `CI` in the `COLLATE` attribute value specifies this. Ensure this variable looks like this: `SQL_Latin1_General_CP1_CI_AS` (not `SQL_Latin1_General_CP1_CS_AS`). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is `perfDB` then the default schema name associated with the user `perfDB` must also be named `perfDB`. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as `sa`. This is because the default schema for the super user is `dbo` and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.

- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.
- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
- f. Click **OK** again to close the Properties window, and then close Component Services.
- g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

2. Configure the JDBC Distributed Transaction Components:

- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
- b. Extract the archive to any folder.
- c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the x64 folder.
- d. Run the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in the master database (for example, for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

Installing and configuring Process Center with an SQL Server database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center.
7. Specify the **Username** and **Password** for this server's administrative account.
 8. Click **Next**.
 9. Select **Yes** to use an existing database.
 10. Specify the required database information.

Table 142. Required database configuration fields for SQL Server

| Field | Action needed |
|---|--|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 1433 or enter the correct server port number. |
| Common database name | Enter the name for the Common database. |
| Process Center database name | Enter the Process Center database name. |
| Performance Data Warehouse database name | Enter the Performance Data Warehouse database name. |
| Common database | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database. |
| Process Center database | For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database. |
| Performance Data Warehouse database | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database. |
| Common Event Infrastructure bus messaging engine database | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine. |
| Service Component Architecture system bus messaging engine database | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine. |

Table 142. Required database configuration fields for SQL Server (continued)

| Field | Action needed |
|--|---|
| Service Component Architecture application bus messaging engine database | <p>For the SCA application bus messaging engine, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine. |

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Center.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 [Installation Manager updates](#)

Installing and configuring Process Server with an SQL Server database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.

- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fix packs, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates
fixpack.2=http://test/rep
launchpad.1=/launchpad_updates
fixpack.WAS_REP=/WAS_updates
fixpack.BPM_REP=/BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.


```
extract_directory/launchpad.sh
```
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Specify the **Username** and **Password** for this server's administrative account.
8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 143. Required database configuration fields for SQL Server

| Field | Action needed |
|---|---|
| Hostname | Accept the default value of localhost or enter the correct database server host name. For example, the IP address. |
| Port | Accept the default value of 1433 or enter the correct server port number. |
| Common database name | Enter the name for the Common database. |
| Process Server database name | Enter the Process Server database name. |
| Performance Data Warehouse database name | Enter the Performance Data Warehouse database name. |
| Common database | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database. |
| Process Server database | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database. |
| Performance Data Warehouse database | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database. |

Table 143. Required database configuration fields for SQL Server (continued)

| Field | Action needed |
|--|--|
| Common Event Infrastructure bus messaging engine database | For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine. |
| Service Component Architecture system bus messaging engine database | For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine. |
| Service Component Architecture application bus messaging engine database | For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine. |

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

Related information:

 Installation Manager updates

Custom installation and configuration:

Use the Custom installation option to install IBM Business Process Manager if you need any installation or configuration options that are not provided by the Typical installation option, if you want to install silently, or if you want to install on an existing installation of WebSphere Application Server.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced:

With custom installation, you can select interactive or silent installation. You must select interactive installation if you want to install on an existing WebSphere Application Server.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced interactively:

You can install the IBM Business Process Manager product package interactively. You must choose interactive installation if you want to install on an existing WebSphere Application Server.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing interactively with a new installation of WebSphere Application Server:

Using custom installation, you can install IBM Business Process Manager with a new installation of WebSphere Application Server.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:

- a. Go to the directory into which you extracted the images.
- b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
5. Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
6. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package.
7. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

8. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
12. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 Installation Manager updates

Installing interactively on an existing installation of WebSphere Application Server on Solaris:

Using custom installation, you can install IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

```
mount_point/launchpad.sh
```
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

```
extract_directory/launchpad.sh
```


- Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
/home/user/bpm_updates.properties
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=/bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=/launchpad_updates  
fixpack.WAS_REP=/WAS_updates  
fixpack.BPM_REP=/BPM_updates
```

- After starting the launchpad, click **Custom installation** on the Welcome page.
- Click **Installation on an existing WebSphere Application Server**.
- Click **Install as administrative user** to install as an administrative user. If you are a root user, you can install as an administrative user. If you are not a root user, or if you want to install to your own user name without root privileges, clear this check box.
- Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
- Click **Install**.
- On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server.
- Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

- On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
- On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
14. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 [Installation Manager updates](#)

Installing IBM Business Process Manager silently:

You can install the IBM Business Process Manager product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing silently using the command line:

You can install IBM Business Process Manager silently using the command line. With this method, you only need to run one command to perform the installation.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager

- WebSphere Application Server Network Deployment, including the ejbdeploy and thinclient features.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

```
extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.

3. Run the following command:

```
extract_directory/IM/tools/imcl install list_of_product_IDs -acceptLicense
-installationDirectory location -repositories repository -showVerboseProgress
-log logName.log
```

where:

- *list_of_product_IDs* is a list of the IDs for the products and features that you want to install. You must include the required features. The syntax is *productID,feature,feature*, with multiple products separated by spaces.

Table 144. Product IDs

| Product | Product ID | Feature and Description |
|------------------|---------------------|--|
| IBM BPM Advanced | com.ibm.bpm.ADV.V80 | <ul style="list-style-type: none"> • bpmAdv.nonprod: Test, staging, or development use. You must specify either this feature or bpmAdv.prod. • bpmAdv.prod: Production use. You must specify either this feature or bpmAdv.nonprod. • wps.client.feature: Required. Run client applications that interact with a IBM BPM Advanced: Process Server in the same cell. • wps.server.feature: Required. Full installation of IBM BPM Advanced: Process Server, which includes the client code and the server code. |

Table 144. Product IDs (continued)

| Product | Product ID | Feature and Description |
|---|--------------------------|---|
| WebSphere Application Server Network Deployment | com.ibm.websphere.ND.v80 | <ul style="list-style-type: none"> core.feature: Required. WebSphere Application Server core content. ejbdeploy: Required. Pre-Enterprise JavaBeans (EJB) 3.0 modules. thinclient: Required. Stand-alone thin clients and resource adapters. embeddablecontainer: Embeddable EJB container. samples: Sample applications feature. com.ibm.sdk.6_32bit: 32-bit software development kit (SDK). You must specify either this feature or com.ibm.sdk.6_64bit. com.ibm.sdk.6_64bit: 64-bit SDK. This feature can only be selected on a 64-bit system. You must specify either this feature or com.ibm.sdk.6_32bit. |
| Installation Manager | com.ibm.cic.agent | <ul style="list-style-type: none"> agent_core: Installation Manager core content. agent_jre: Installation Manager Java Runtime Environment (JRE). |

- *location* is the path to the directory where you want to install the products.
- *repository* is the path to the repository where you have extracted the files, one of the following directories:

extract_directory/repos_32bit
extract_directory/repos_64bit

For more than one repository, separate the repository locations with commas.

- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas. Create encrypted passwords using the IBM Installation Manager.

Table 145. Keys

| Key | Description |
|-----------------------|---|
| user.db2.use.existing | <p>If you are using an existing DB2 setup, add the following line exactly:</p> <pre>user.db2.use.existing=true</pre> <p>If DB2 must be installed, specify the value <i>false</i>.</p> |
| user.db2.port | Port for the DB2 database. The default value is 50000. |

- *logName* is the name of the log file to record messages and results.

Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for imcl.

Installation Manager installs the list of products and writes a log file to the directory that you specified. The log file is empty if there are no errors or warnings.

The following example installs IBM Business Process Manager Advanced and WebSphere Application Server Network Deployment on Solaris.

```
imcl install com.ibm.bpm.ADV.v80,bpmAdv.prod,wps.client.feature,wps.server.feature
com.ibm.websphere.ND.v80,core.feature,ejbdeploy,thinclient,embeddablecontainer,samples,com.ibm.sdk.6_32bit
-acceptLicense -installationDirectory /usr/IBM/BPM80 -repositories /usr/tmp/BPM/repository
-showVerboseProgress -log silentinstall.log
```

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

Related reference:

 Command-line arguments for imcl

 IBM Business Process Manager Advanced system requirements

Related information:

 IBM WebSphere Application Server Information Center

Installing silently using a response file:

You can install IBM Business Process Manager Advanced silently by creating a response file and then running a command to use that response file to install the product.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

By using response files, you can simplify the silent installation and reduce error in the process because you set up your installation options once in a saved, sharable file that can be used on one or more machines. The installation software provides sample response files for each supported operating system and bit version. You can use an unmodified sample response file to perform a silent installation using default settings, or you can modify the response file to set particular values. The comments in the sample response files provide detailed instructions and information about setting the values.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.
`extract_directory/IM/tools/imutilsc -silent -nosplash encryptString password_to_encrypt`
2. Create the response file that will install the required base products and IBM Business Process Manager Advanced. Copy a sample response file, suitable for your bit version and user access level, from the following directory:

```
extract_directory/responsefiles/BPM/
```

Alternatively, you can create a response file by recording your actions in Installation Manager. When you record a response file, the selections that you make in Installation Manager are stored in an XML file. When you run Installation Manager in silent mode, Installation Manager uses the data in the XML response file to perform the installation.

3. The default values provided in the sample response files will perform a basic install, but you should review the file and its comments, and modify the parameters as needed for your environment and the access level of your user ID.

In particular, review the following parameters and values:

- For non-root user IDs, check that all location variables point to locations that the user has permissions for.
 - Repository location. If you are not running directly from the *extract_directory/responsefiles/BPM/* directory, point to the location of your installation repository. The repository can be local or remote. If you are running from the DVD, copy the response file from the DVD and point it back at the repository on the DVD.
 - Installation location (**installLocation**) where IBM Installation Manager is already installed or will be installed.
 - Installation location where IBM Business Process Manager Advanced will be installed.
 - Eclipse location (**eclipseLocation**).
 - The list of features for the product.
 - Production or non-production use.
4. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.
 5. Run the following command:

Root user:

```
extract_directory/IM/installc -acceptLicense input  
extract_directory/responsefiles/BPM/response_file_name.xml  
-log preferred_log_location/silent_install.log
```

Nonroot user:


```
extract_directory/IM/userinstc -acceptLicense input  
extract_directory/responsefiles/BPM/response_file_name.xml  
-log preferred_log_location/silent_install.log
```

Installation Manager installs any required prerequisites and IBM Business Process Manager Advanced, and writes a log file to the directory you specified.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

Related tasks:

 [Installing silently by using Installation Manager](#)

 [Recording a response file with Installation Manager](#)

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Granting write permission of files and directories to nonroot users for profile creation or augmentation:

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root/profiles* directory of the product.

Restrictions:

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
 - a. Create a group named **profilers**, which will contain all users who can create profiles.
 - b. Create a user named **user1**, who can create profiles.
 - c. Add users **product_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
 - Create the *install_root/logs/manageprofiles* directory:
`mkdir install_root/logs/manageprofiles`
 - Create the *install_root/properties/fsdb* directory:
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the *profileRegistry.xml* file as the installer. For this example, the file path is:
`install_root/properties/profileRegistry.xml`

Add the following information to the *profileRegistry.xml* file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **/opt/IBM/WebSphere/AppServer**.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

The root user and the nonroot user can use the same tasks to manage profiles.

Configuring profiles and databases:

After you install the product, if you did not use typical installation, you must create or augment one or more profiles to define the runtime environment. Before starting a profile, you must have configured the databases that are to be used with the profile.

Configuring a stand-alone environment with a DB2 database server on Solaris:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 database server.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECON 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run `vi /etc/profile`

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run `vi /etc/group`

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103:db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

Creating database design files for DB2:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5)wesb.nd.topology
(6)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
```

```
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
```

```

(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibME] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibME] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]

```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

```

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]

```

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.standalone is complete with 0 remaining item(s):
```

- When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

- Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.

- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```



```
-v db_design_file | db_scripts_output_directory
  when a db_design_file is given, validation will be done on the specified
  database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
  generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :

(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Database User name[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

Regular pagesize[default=32k] :

Regular TableSpace[default=TWSSPACE] :

Temporary pagesize[default=32k] :

Temporary TableSpace[default=TWSTEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with a DB2 database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the DB2 database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

If you plan to configure the databases on a remote server, you must have created the databases before beginning to create or augment the profile. You can create databases on the local server or use existing databases on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run
`vi /etc/profile`

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH  
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run
`vi /etc/group`

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102;root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:
`/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")`
- When the DB2 database manager is not running:
`SQL1032N No start database manager command was issued. SQLSTATE=57019`
- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:
`SQL1092N "ROOT" does not have the authority to perform the requested command.`

Creating Process Center stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the `install_root/.Xdefaults` file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.

- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.
 - e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step
15. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 146. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next** .
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can

include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.

d. Click **Next**.

14. On the Process Server Configuration page, set the values for the following parameters.

- **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
- **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. Advanced: Configure the databases using a design file.

- a. Select **Use a database design file**.
- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.
- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 147. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.

22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- **PC_Std_DMGr_DB2.response**
- **PC_Std_DMGr_DB2zOS.response**
- **PC_Std_DMGr_Oracle.response**
- **PC_Std_DMGr_SQLServer.response**
- **PC_Adv_DMGr_DB2.response**
- **PC_Adv_DMGr_DB2zOS.response**
- **PC_Adv_DMGr_Oracle.response**
- **PC_Adv_DMGr_SQLServer.response**
- **PC_Std_Managed_DB2.response**

- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2z0S.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2z0S.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
```

```

# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies

```

```

#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####

```

```

# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies

```



```

# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbHostName
#
# Description:
# The database server host name or IP address. The default value is the
# local host name. Required when the -bpmdbDesign parameter was not
# specified. (It is not required if the -bpmdbDesign parameter is
# specified.)
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbServerPort

```

```

#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword

```

```

#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter:  cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter:  dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#

```



```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false

```

```

# true
# Default Values:
# false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
# Determines whether the Business Process Choreographer sample configuration
# is created.
# The following conditions apply:
# Only DB2 databases are supported.
# The default is set to true for DB2.
# It uses its own schema within the common database (CMNDB) and the schema
# name is always BPEDB.
# Business Process Choreographer database parameters set in the Database
# Design Tool and generated to the database design file are ignored by
# Business Process Choreographer at profile creation time.
# See -bpmdbDesign for information about the database design file.
# Note: If these conditions are too restrictive, consider setting
# -configureBPC to false and using other documented methods to configure
# Business Process Choreographer.
#
# Valid Values:
# false
# true
# Default Values:
# true
#####
configureBPC=true

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2.response
#

```

```

# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

```

```

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:

```

```

# Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period

```

```

#keyStorePassword =keyStore_password

#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or

```



```

# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword

```

```

#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

```

```

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port

```

```

# number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the

```

```

# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMPassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
# The schema name for the Common database. If no value is specified, the
# following default database schema names are used:
# DB2 database
# The default schema name is the value specified by the dbUserId parameter.
# The -cdbSchemaName parameter is valid for all database types except for
# Oracle and Microsoft SQL Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType

```



```

#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to

```

```

# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,

```

```

# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,

```

```

# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId

```

processCenterPassword=process_Center_Password

```
#####  
# Parameter: enableAdminSecurity  
#  
# Description:  
#   For IBM Business Process Manager omit this parameter. Administrative  
#   security is always enabled for IBM Business Process Manager profiles.  
#   You must also specify the parameters -adminUserName and -adminPassword  
#   along with the values for these parameters.  
#  
# Valid Values:  
#   true  
# Default Values:  
#   true  
#####  
enableAdminSecurity=true  
adminUserName=adminUser_ID  
adminPassword=adminPassword  
  
#####  
# Parameter: signingCertDN  
#  
# Description:  
#   Specifies the distinguished name of the root signing certificate that you  
#   create when you create the profile. Specify the distinguished name in  
#   quotation marks. This default personal certificate is located in the  
#   server keystore file. If you do not specifically create or import a root  
#   signing certificate, one is created by default. See the  
#   -signingCertValidityPeriod parameter and the -keyStorePassword.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: signingCertValidityPeriod  
#  
# Description:  
#   An optional parameter that specifies the amount of time in years that the  
#   root signing certificate is valid. If you do not specify this parameter  
#   with the -signingCertDN parameter, the root signing certificate is valid  
#   for 20 years.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: keyStorePassword  
#  
# Description:  
#   Specifies the password to use on all keystore files created during profile  
#   creation. Keystore files are created for the default personal certificate  
#   and the root signing certificate.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#####  
#signingCertDN =distinguished_name  
#signingCertValidityPeriod =validity_period  
#keyStorePassword =keyStore_password  
  
#####  
# Parameter: defaultPorts
```

```

#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the

```

```

#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.

```



```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:

```

```

# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#    Database name for Process Server database.  The value specified for
#    -procSvrDbName and -perfDWDbName must be different.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#    The Process Server user name.  Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#    The Process Server password.  Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#    Database name for Performance Data Warehouse.  The value specified for
#    -perfDWDbName and -procSvrDbName must be different.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#    Performance Data Warehouse user name.  Required for setting up the
#    Performance Data Warehouse.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

```

```

#####
# Parameter:  cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
# Parameter:  dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID

```

```

#dbSysMePassword=sys_me_pwd

#####
# Parameter:  environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```



```

#####
# Parameter: webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC

```

```

#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#####
configureBPC=true

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create

```

```

#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.

```

```

#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL

```

```

#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you

```

```

# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.

```

```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual

```



```

#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#    Specifies the file that holds the database configuration design for all of
#    the IBM Business Process Manager components .
#    This parameter replaces the -wbldbDesign that was used in previous
#    versions of WebSphere Process Server.
#
#    Note: If a bpmdbDesign file parameter is specified, none of the database
#    related properties should be specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#bpmdbDesign=design_file
#####
# Parameter:  dbCreateNew
#
# Description:
#    Indicates a new database is created or if an existing database is to be
#    reused. Valid values are true or false. This value cannot be true for an
#    Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#    a DB2 database.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#####
dbCreateNew=true
#####
# Parameter:  dbType
#
# Description:
#    The database type. Set one of the following values for the type of
#    database product you are using with IBM Business Process Manager.
#
#    Required only for one of the following:
#    A stand-alone profile when -bpmdbDesign was not specified.
#    An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false

```

```

#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:

```

```

# The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

```

```
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Augmenting stand-alone profiles for Process Center with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.
- Skip to the Business Space Configuration step
9. On the Database Configuration - Part 1 page, specify your database information.

- a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 148. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.

- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy

server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 149. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/db2/java.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.

Table 149. Required database configuration fields for DB2 Database (continued)

Field	Action needed
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

15. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
16. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - *default.procctr*: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - *default.procctr.adv*: for a Process Center stand-alone profile for IBM BPM Advanced.
 - *default.procsvr*: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - *default.procsvr.adv*: for a Process Server stand-alone profile for IBM BPM Advanced.
 - *default.esbserver*: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the *augment* parameter to make changes to an existing profile with an augmentation template. The *augment* parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- *PC_Std_DMGr_DB2.response*
- *PC_Std_DMGr_DB2z0S.response*
- *PC_Std_DMGr_Oracle.response*
- *PC_Std_DMGr_SQLServer.response*
- *PC_Adv_DMGr_DB2.response*
- *PC_Adv_DMGr_DB2z0S.response*
- *PC_Adv_DMGr_Oracle.response*
- *PC_Adv_DMGr_SQLServer.response*
- *PC_Std_Managed_DB2.response*
- *PC_Std_Managed_DB2z0S.response*
- *PC_Std_Managed_Oracle.response*
- *PC_Std_Managed_SQLServer.response*
- *PC_Adv_Managed_DB2.response*
- *PC_Adv_Managed_DB2z0S.response*
- *PC_Adv_Managed_Oracle.response*
- *PC_Adv_Managed_SQLServer.response*
- *PC_Std_StandAlone_DB2.response*
- *PC_Std_StandAlone_DB2z0S.response*

- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
 -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring DB2 databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/ProcessServer/DB2/* at profile creation time.
2. Run the following script.

```
configProcessServerDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.

- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/PerformanceDW/DB2/* at profile creation time.
 2. Run the following script.

```
configPerformanceDWDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
 - Chosen not to run the database scripts to initialize the Common database during the configuration.
 - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/DB2/*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

configCommonDB.sh createDB - creates the database and tables

configCommonDB.sh - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is `install_root/profile/dbscripts/BusinessSpace/node_name_server_name/DB2/CMNDB`.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:
configBusinessSpaceDB.sh Connect to the database, bind the Command Line Interface to the Business Space database using the **db2 bind DB2_installation_directory\bnd\@db2cli.lst blocking all grant public** command, and then reset the database connection.

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.timestamp.log`. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Configuring a stand-alone environment with a DB2 for z/OS database server on Solaris:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

Creating database design files and database scripts (Solaris stand-alone):

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS:

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory `/tmp/DB2-zOS/W1MYDB`. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as `/tmp/DB2-zOS/W8CELLDB`. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as `/tmp/DB2-zOS/W8S1DB`. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the `[begin] = BSpace : WBI_BSPACE` section, changing the `databaseName` value to `BSPACE`.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the `DbDesignGenerator.sh` command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2) bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI_BPC : [status = not complete]
- (3) [BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4) [BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI_BSPACE : [status = not complete]
- (6) [SibME] WBI_BPC_ME : [status = not complete]
- (7) [SibME] WBI_CEI_ME : [status = not complete]
- (8) [SibME] WBI_SCA_APP_ME : [status = not complete]
- (9) [SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10) [SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11) [SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type `y` and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for `WBI_CommonDB` [default=] :

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, `WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB`. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type `2` to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter `'s'`; or enter anything else to continue :

9. To configure the data source properties component, type anything other than `s` and press Enter. To skip this configuration and accept the defaults, type `s` and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

Note: For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

```
[info] Please pick one of the following [Is this database for a Process Center?(s)] :
```

- (1) false
- (2) true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.standalone**, after you configure the master component, you must manually configure the **BPM_PerformanceDW** and **BPM_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

```
Please enter the number for the database component :12
```

```
[status] bpm.advanced.standalone.topology is not complete with 6 remaining item(s):  
[ 1 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property  
'schemaName' for SCHEMA is empty.  
[ 2 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property  
'catalog' for VCAT is empty.  
[ 3 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property  
'adminUserName' for SECURITY_ADMIN_USER is empty.  
[ 4 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property  
'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.  
[ 5 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property  
'schemaName' for SCHEMA is empty.  
[ 6 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property  
'catalog' for VCAT is empty.
```


11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```
12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```
13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```
14. To generate the database scripts for each component in the directory that you specified earlier, type **y** and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB

After the last set of scripts is generated, you see information similar to the following example:
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment:

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:

- The location of the database tables.
- The location of the JDBC driver on the system where the server profile will be created.
- The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :
```

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm_performancedw
- (4) bpm_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 2 to select **(2) DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Data source user name[default=] :

Database schema name (SQLID)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1) false

(2>true

Please enter the number for the Is this database for a Process Center? [default=false] :
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS  
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database Location on z/OS system[default=] :  
Database server host[default=] :  
Database server port[default=] :  
Data source user name[default=] :  
Data source password[default=] :  
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
10. To generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Related tasks:

“Creating the messaging engine data stores” on page 2052

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Generating scripts for creating database objects in DB2 for z/OS:

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
 - Database server details
 - Data source details
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database

design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory `/tmp/DB2-zOS/W1MYDB`. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as `/tmp/DB2-zOS/W8CELLDB`. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
 - To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as `/tmp/DB2-zOS/W8S1DB`. You can later run the generated schema to create the database objects for these components in the W8S1DB database.
1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
 2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: `cd /opt/IBM/WebSphere/AppServer/util/dbUtils`

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db_design_file* is the fully qualified name of the database design file. For example:


```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

Tip: In the database design file, the directories for the SQL scripts are defined by the `ddl_outDir` variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the `ddl_outDir` variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

Note: The **createDB2.sh** script requires execute permissions.

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
```

```
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with a DB2 for z/OS database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Creating Process Center stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the `install_root/.Xdefaults` file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.

7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**.
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web

server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.

- d. Click **Next**.
14. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating Process Server stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.

- b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response

- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT
```

```

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses DB2z0S.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2z0S.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere

```

```

# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01
#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard
#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.

```

```

# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:

```

```

# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value

```

```

#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbldbDesign that was used in previous
#     versions of WebSphere Process Server.

```



```

#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true
#####
# Parameter: dbName

```

```

#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter:  dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false

```

```

# true
# Default Values:
# false
#####
configureBRM=false

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#

```

```

# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)

```



```

#     shortHostNameCellCellNumber
#     else
#     shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#

```

```

# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port

```

```

# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter:   winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:   winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:   winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:   winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:   winserviceStartupType
#
# Description:

```

```

# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2UDBOS390
# Default Values:
# DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword

```

```

#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing

```



```

# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.

```

```

#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2z0S.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#    Set a performance tuning level appropriate for the profile you are
#    creating.
#    This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#    None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#    Specifies the cell name of the profile. Use a unique cell name for each
#    profile.
#    The default value for this parameter is based on a combination of the
#    short host name, the constant Cell, and a trailing number, for example:
#    if (DMgr)
#        shortHostNameCellCellNumber
#    else
#        shortHostNameNodeNodeNumberCell
#    where CellNumber is a sequential number starting at 01 and NodeNumber is
#    the node number that you used to define the node name.
#    The value for this parameter must not contain spaces or any characters
#    that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter: nodeName
#
# Description:
#    Specifies the node name for the node that is created with the new profile.
#    Use a unique value within the cell or on the workstation. Each profile
#    that shares the same set of product binaries must have a unique node name.
#    This parameter is required for profile creation only with the
#    dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#    procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#    The default value for this parameter is based on the short host name,
#    profile type, and a trailing number, for example:
#    if (DMgr)
#        shortHostNameCellManagerNodeNumber
#    else
#        shortHostNameNodeNodeNumber
#    where NodeNumber is a sequential number starting at 01.
#    The value for this parameter must not contain spaces or any characters
#    that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:

```

```

# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true

```

```

# Default Values:
#   None
#
# Parameter:  serviceName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this

```

```

#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbldbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
#bpmdbDesign=design_file

#####
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
#####
dbCreateNew=false

```



```

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbStorageGroup
#
# Description:
# The storage group name for DB2 z/OS databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbConnectionLocation
#
# Description:
# The location of DB2 for z/OS database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#

```

```

# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
isDeveloperServer=false

#####
# Parameter:  webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false

```

```

#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#

```

```

# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the

```

```

# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
# Set a performance tuning level appropriate for the profile you are
# creating.
# This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
# development
# standard
# production
# Default Values:
# None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
# shortHostNameCellCellNumber
# else
# shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
# shortHostNameCellManagerNodeNumber
# else

```

```

#           shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter:  enableAdminSecurity

```



```

#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.

```

```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbiddbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.

```

```

#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database

```

```

# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

```

```

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including *tw_admin* and *tw_user*. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.

- d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
9. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
10. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
11. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
12. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Augmenting stand-alone profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

- Advanced: Configure the databases using a design file.
 - Select **Use a database design file**.
 - Click **Browse**.
 - Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - Click **Next**. The design file location is passed to the command line to complete the database configuration.
- Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
- Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
- On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
- On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

- Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
- Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.

- `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2zOS.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2zOS.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2zOS.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2zOS.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2zOS.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2zOS.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2zOS.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`
- `PS_Adv_DMGr_DB2.response`

- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Creating and configuring DB2 for z/OS databases after stand-alone profile creation:

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

DB2 for z/OS authorization prerequisites:

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as #DB_USER#. These permissions are available in the createDatabase.sql database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;  
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;  
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the #DB_USER# user to create sequences and stored procedures with a schema qualifier of #SCHEMA#:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;  
GRANT BINDADD TO #DB_USER#;
```

Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database #DB_NAME# can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
```

```

GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;

```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named SICELL, and connect user WSADMIN to it, as follows:

```

INFORMATION FOR GROUP SICELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
WSADMIN CONNECT 000000 NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE RESUME DATE=NONE

```

Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the

system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating databases in the DB2 for z/OS subsystem:

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

Considerations for choosing the `createDB2.sh` script

- `createDB2.sh` can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- `createDB2.sh` runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- `createDB2.sh` runs the SQL for each component in the correct sequence.
- `createDB2.sh` creates database objects according to a naming convention that you define.
- `createDB2.sh` organizes the layout of database objects across DB2 for z/OS databases.
- `createDB2.sh` issues GRANT permissions to database, storage group, and buffer pool objects.
- `createDB2.sh` automatically completes customization of the service integration bus Data Definition Language (DDL).
- `createDB2.sh` runs in a UNIX System Services environment.
- `createDB2.sh` produces an audit trail of the objects that it creates.

Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the `createDB2.sh` script in a mode that bypasses execution of the SQL statements and instead consolidates them into two `.sql` files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

Related tasks:

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Configuring the DB2 command line processor:

Before you run the `createDB2.sh` script, you must configure the DB2 command line processor by defining a set of environment variables and a `db2` command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `c1p.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the `createDB2.sh` script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
 - Modify the CLASSPATH environment variable to include the `c1p.jar` file.
 - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.

- Define the **db2** command as an alias for the command that starts the command line processor.
- Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=
/file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=
/wasv8config/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:
 - A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: `server:port/database`. The port is optional, and the DB2 location name must be specified in uppercase characters.
 - A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

Tip: When you define a `DB2ALIASNAME` value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPF panel and update the DBADM CREATE AUTH field to set `DB2 ZPARAM DBACRVW=YES`.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two `.sql` files that you can run later using other database tools.

Related tasks:

“Creating DB2 for z/OS database objects using the `createDB2.sh` script” on page 2047

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2048

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

Creating DB2 for z/OS database objects using the `createDB2.sh` script:

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate  
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

-RunSQL

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP -DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

Tip: The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 2045

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using the DB2 command line processor:

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can

run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate  
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

Tip: By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two `.sql` files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

The `z_schema.sql` and `z_schemaProc.sql` files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The `z_schemaProc.sql` file contains SQL statements for stored procedures, and is generated in addition to `z_schema.sql` because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the `z_schema.sql` file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the `z_schemaProc.sql` file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 2045

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

Tip: The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:


```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

Tip: If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

```
@STOGRP@    = DB2 Storage group name
@DB_NAME@   = DB2 Database name
@SCHEMA@    = DB2 SQLID (Schema Qualifier)
@BPTABLE4K@ = Buffer Pool of 4k Size
@BPINDEX@   = Buffer Pool Index
@VOLUMES@   = DB2 comma separated list of volume-ids
@VCAT@      = DB2 integrated catalog facility name
@DB_USER@   = The user ID which will be used to create the database objects
```

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.

5. Run the customized scripts by using the tool of your choice.

Tip: If you previously ran the `createDB2.sh` script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (`z_schema.sql` and `z_schemaProc.sql`), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

Related tasks:

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2048

When you run the `createDB2.sh` script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the `createDB2.sh` script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

Creating the messaging engine data stores:

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
- When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
 - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
 - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.

1. For each messaging engine, complete the following steps to generate the required database scripts:
 - a. Run the database design tool (**DbDesignGenerator**) from the `install_root/util/dbUtils` directory.
 - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
 - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

Tip: As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for**

Standalone profile or Deployment Environment. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

Related tasks:

“Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment” on page 1974

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS” on page 1968

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

Note: Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Setting the correct schema name for the messaging engines:

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
 - a. Select **Messaging engines**, and then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Verify that the messaging engines can be started:
 - a. Log out of the administrative console.
 - b. Stop, and then restart the server.
 - c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"  
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

Configuring a stand-alone environment with an Oracle database server on Solaris:

You can configure a stand-alone environment for IBM Business Process Manager using an Oracle database server.

Setting up Oracle databases:

You or your database administrator (DBA) can configure the Process Server database, Performance Data Warehouse database, and Common database before, after, or during creating or augmenting a profile.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

You have the following options for creating and configuring the databases:

- You or your DBA creates the databases in advance and the Profile Management Tool configures them for you during profile creation or augmentation.
- You or your DBA creates a database design file using the database design tool, and you pass the file to the Profile Management Tool or **manageprofiles** command-line utility during profile creation or augmentation.
- You postpone the database configuration, and you or your DBA creates the required databases and database tables later by running the scripts that are generated during profile creation or augmentation.

Creating database design files for Oracle:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with Oracle:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
```

- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2) bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.
 [info] Completed database components can be edited to change existing or defaulted property values.
 [info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI_BPC : [status = not complete]
- (3) [BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4) [BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI_BSPACE : [status = not complete]
- (6) [SibME] WBI_BPC_ME : [status = not complete]
- (7) [SibME] WBI_CEI_ME : [status = not complete]
- (8) [SibME] WBI_SCA_APP_ME : [status = not complete]
- (9) [SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10) [SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11) [SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI_CommonDB is not complete with 1 remaining item(s):
 [1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool

is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

- (1)oci8
- (2)thin

Please enter the number for the Oracle driver type(s)? [default=thin] :
 Data source user name[default=] :
 Data source password[default=] :
 Oracle JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI_CommonDB is complete with 0 remaining item(s):

```
-----
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.standalone**, you must manually configure the **BPM_PerformanceDW**, **BPM_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]


```
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

Please enter the number for the database component :3

```
-----
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[Bspace] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

Please enter the number for the database component :4

```
-----
[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[ 1 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
[ 2 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[ 3 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with Oracle in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :

(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
```

- (6)cei
- (7)sca
- (8)sibme
- (9>wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=BPMDB] :

Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
```

```
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle  
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
WAS_HOME/profiles/default/util/dbUtils/Oracle-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with an Oracle database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the Oracle database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

To configure the database information while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:


```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating Process Center stand-alone profiles with an Oracle database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.

- Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 150. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.

Table 150. Required database configuration fields for Oracle (continued)

Field	Action needed
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with an Oracle database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name**: The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type**: Select how the Process Server is used:

- Select **Production** if the server is to be used in a production capacity.
- Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
- Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

d. Click **Next**.

17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 151. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>

Table 151. Required database configuration fields for Oracle (continued)

Field	Action needed
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.

- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
 - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1'** or **'winserviceCheck=false'**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response

- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters.
For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
```

```

# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####

```

```

# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####

# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.
#####

```



```
cellName=cell_name
nodeName=node_name
serverName=server1
```

```
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
```

```

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option

```

```

# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:

```

```

# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew

```

```

#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:

```

```

# The directory of the ORACLE_HOME directory. This parameter is required
# when the parameter dbDelayConfig is set to false. No, this parameter is
# only valid if you are using Oracle databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=orc1
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.

```



```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine. This
# parameter is required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

```

```

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbAppMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
#   user-specified a user name and password during profile creation, this
#   parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword. If you enter a user-specified a user name and password during
#   profile creation, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:

```

```

# false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: webFormInstallRoot
#
# Description:
# Specifies the full path where IBM Forms Server is installed. For example,
# the IBM Forms Server install root might be
# C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
# Configures the business process rules manager. The default value is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
configureBRM=false

```

```
#####
# Parameter:  configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
configureBPC=false
```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####
```

```

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are

```

```

#    creating.
#    This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#    None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#    Specifies the cell name of the profile. Use a unique cell name for each
#    profile.
#    The default value for this parameter is based on a combination of the
#    short host name, the constant Cell, and a trailing number, for example:
#    if (DMgr)
#        shortHostNameCellCellNumber
#    else
#        shortHostNameNodeNodeNumberCell
#    where CellNumber is a sequential number starting at 01 and NodeNumber is
#    the node number that you used to define the node name.
#    The value for this parameter must not contain spaces or any characters
#    that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#    Specifies the node name for the node that is created with the new profile.
#    Use a unique value within the cell or on the workstation. Each profile
#    that shares the same set of product binaries must have a unique node name.
#    This parameter is required for profile creation only with the
#    dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#    procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#    The default value for this parameter is based on the short host name,
#    profile type, and a trailing number, for example:
#    if (DMgr)
#        shortHostNameCellManagerNodeNumber
#    else
#        shortHostNameNodeNodeNumber
#    where NodeNumber is a sequential number starting at 01.
#    The value for this parameter must not contain spaces or any characters
#    that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
# Description:
#    Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

```



```

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:

```

```

# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user

```

```

# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter:  dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an

```

```

# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:
# The directory of the ORACLE_HOME directory. This parameter is required
# when the parameter dbDelayConfig is set to false. No, this parameter is
# only valid if you are using Oracle databases.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database.  The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse.  The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name.  Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword

```

```

#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine. This
#   parameter is required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:

```



```

# The user id to authenticate with the Common database. For Oracle, the
# default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
# Process Manager profiles and advanced deployment manager (dmgr.*.adv)
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
# and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.

```

```

# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on

```

```

# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

```

```

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.

```

```

#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID

```

adminPassword=adminPassword

```
#####  
# Parameter: signingCertDN  
#  
# Description:  
#   Specifies the distinguished name of the root signing certificate that you  
#   create when you create the profile. Specify the distinguished name in  
#   quotation marks. This default personal certificate is located in the  
#   server keystore file. If you do not specifically create or import a root  
#   signing certificate, one is created by default. See the  
#   -signingCertValidityPeriod parameter and the -keyStorePassword.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: signingCertValidityPeriod  
#  
# Description:  
#   An optional parameter that specifies the amount of time in years that the  
#   root signing certificate is valid. If you do not specify this parameter  
#   with the -signingCertDN parameter, the root signing certificate is valid  
#   for 20 years.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: keyStorePassword  
#  
# Description:  
#   Specifies the password to use on all keystore files created during profile  
#   creation. Keystore files are created for the default personal certificate  
#   and the root signing certificate.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#####  
#signingCertDN =distinguished_name  
#signingCertValidityPeriod =validity_period  
#keyStorePassword =keyStore_password  
#####  
# Parameter: defaultPorts  
#  
# Description:  
#   Assigns the default or base port values to the profile.d o not use this  
#   parameter when using the -startingPort or -portsFile parameter.  
#   During profile creation, the manageprofiles command-line utility uses an  
#   automatically generated set of recommended ports if you do not specify the  
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile  
#   parameter. The recommended port values can be different than the default  
#   port values based on the availability of the default ports.  
#  
# Valid Values:  
#   Varies  
# Default Values:  
#   None  
#  
# Parameter: startingPort  
#  
# Description:
```

```

# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:

```

```

# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:

```



```

# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.

```

```

#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter: dbLocation
#
# Description:
# The directory of the ORACLE_HOME directory. This parameter is required
# when the parameter dbDelayConfig is set to false. No, this parameter is
# only valid if you are using Oracle databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbLocation=db_location
#####
# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the

```

```

# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# orcl
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
# Database name for Process Server database. The value specified for
# -procSvrDbName and -perfDWDbName can be the same.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the

```

```

# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine. This
# parameter is required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database. For Oracle, the
# default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
# Process Manager profiles and advanced deployment manager (dmgr.*.adv)
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
# and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword

```

```

#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbAppMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd
#####
# Parameter: dbCeiMeUserId
#
# Description:
#   The default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
#   password during profile creation and if you specified ORACLE for dbType,
#   this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd
#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
#   user-specified a user name and password during profile creation, this
#   parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword. If you enter a user-specified a user name and password during
#   profile creation, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####

```

```

# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter:  webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true

```



```

# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#

```

```

# Valid Values:
#   false
# Default Values:
#   false
#####
configureBPC=false

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.

```

```

#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:

```

```

#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process

```

```

# Center). It is required for a stand-alone process server (standard or
# advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.

```

```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName

```

```

#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```



```

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   orcl
#

```

```

# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName can be the same.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#    Database name for Performance Data Warehouse.  The value specified for
#    -perfDWDbName and -procSvrDbName can be the same.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#    Performance Data Warehouse user name.  Required for setting up the
#    Performance Data Warehouse.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPASSWORD
#
# Description:
#    Performance Data Warehouse password.  Required for setting up the
#    Performance Data Warehouse.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter:  dbProcSvrMeUserId
#
# Description:
#    Process Server Messaging Engine user name.  Required for setting up the
#    Process Server Messaging Engine.  If you enter a user-specified a user name
#    and password during profile creation and if you specified ORACLE for
#    dbType, this parameter is required.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####

```

```

# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine. This
# parameter is required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database. For Oracle, the
# default user name is the first three characters of the Oracle database
# service name. For example: If the database service name is ORCL,
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
# Process Manager profiles and advanced deployment manager (dmgr.*.adv)
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
# and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter. Required for
# all stand-alone IBM Business Process Manager profiles and advanced
# deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
# to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#    Specifies the environment type. The environment type refers to how IBM
#    Business Process Manager is used; for example, in a production, stage or
#    test environment. Load testing might be done on a test server, while a
#    stage environment type might be used as a temporary location to host
#    changes before putting those changes into production. You might specify
#    Stage as the Environment type if the server you are configuring will be
#    accessed and used to review content and new functionality. Valid values
#    are as follows:
#    Test
#    Use Test if the server you are configuring is to be used as a testing
#    environment.
#    Stage
#    Use Stage if the server is to serve as a staging platform to be used as a
#    preproduction server.
#    Production
#    Use Production if the server is to serve in a production capacity.
#    The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#    Specifies whether the server is intended for development purposes only.
#    This parameter is useful when creating profiles to test applications on a
#    non-production server prior to deploying the applications on their
#    production application servers. If -isDeveloperServer is set when
#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with an Oracle database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
8. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine

- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 152. Required database configuration fields for Oracle

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <p>User name Enter a user name for the Common database (such as CMNDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p>
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
 - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
 - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 13. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
 14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with an Oracle database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.

- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.
 - e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

Note: You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 153. Required database configuration fields for Oracle

Field	Action needed
Common database	For the Common database, enter values for the following parameters: User name Enter a user name for the Common database (such as CMNDBA). Password Enter the password that you specified when you created the user in Oracle. Confirm password Confirm the password by reentering it.

Table 153. Required database configuration fields for Oracle (continued)

Field	Action needed
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter a user name for the Process Server database (such as BPMDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p>Password Enter the password that you specified when you created the user in Oracle.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\oracle

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
- When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
- The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions

are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response

- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2z0S.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2z0S.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring Oracle databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/ProcessServer/Oracle/* at profile creation time.
2. Run the following script.

```
configProcessServerDB.sh
```

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/PerformanceDW/Oracle/* at profile creation time.
2. Run the following script.

```
configPerformanceDWDB.sh
```

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the Common database during the configuration.
- Created the database that you want to create the Common database tables in.

1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/Oracle/*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run the following command:

configCommonDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install_root/profile/dbscripts/BusinessSpace/node_name_server_name/Oracle/CMNDB*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:
configBusinessSpaceDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:
 - a. **createDatabase.sql**
 - b. **createTable_BusinessSpace.sql**
 - c. **createGrant_BusinessSpace.sql**

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Configuring a stand-alone environment with an SQL Server database server on Solaris:

You can configure a stand-alone environment for IBM Business Process Manager using a Microsoft SQL Server database server.

Creating SQL Server databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password  
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database,

and `database_name` is the name of the database that you are creating. `COLLATE SQL_Latin1_General_CP1_CI_AS` is included because these databases must be case-insensitive.

b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the `COLLATE` clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters `CI` in the `COLLATE` attribute value specifies this. Ensure this variable looks like this: `SQL_Latin1_General_CP1_CI_AS` (not `SQL_Latin1_General_CP1_CS_AS`). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmntExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is `perfDB` then the default schema name associated with the user `perfDB` must also be named `perfDB`. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as `sa`. This is because the default schema for the super user is `dbo` and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- From the Design view, press F4 to view the Properties window.
- From the Properties window, update the schema name.
- Right-click the tab and select **Close** to close the Design view.

- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Creating database design files for SQL Server:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with SQL Server:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
```


generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5)wesb.nd.topology
(6)wesb.standalone
```

- To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

- Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

- To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

- Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

- Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database server host[default=] :
Database User name[default=] :
Database schema[default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :
```

```
[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :
```

```
(1) false
(2) true
```

```
Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI_CommonDB is complete with 0 remaining item(s):

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the **BPM_ProcessServer**, you must manually configure the **BPM_PerformanceDW** component for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
- (2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (5)[save and exit]

Please enter the number for the database component :2

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with SQL Server in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
```

- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm_performancedw
- (4)bpm_processserver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Database server host[default=] :

Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

- At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

- To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)
```

- Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :
```

```
[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :
```

```
(1) false
(2) true
```

```
Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :
```

Note: The password is encrypted in the generated output files.

- At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

- Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```


10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type `y` and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/SQLServer-ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'!"..**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
 - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
 - c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
 - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating or augmenting stand-alone profiles with an SQL Server database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the SQL Server database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Restriction: The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 154.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner dbo using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDBE00 (Business Process Choreographer Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

IBM Business Process Manager packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the Detailed hardware and software requirements for IBM Business Process Manager page.

Note: You are responsible for providing JDBC driver levels outside of what is packaged with IBM Business Process Manager.

You must configure XA transactions after the database is installed and before you start the server.

Creating Process Center stand-alone profiles with an SQL Server database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next** .
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.

- b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 155. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.

Table 155. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 156. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
 - **Windows Authentication:** With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
17. **Advanced:** On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 18. **Advanced:** Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with an SQL Server database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next** .
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.

- b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
14. On the Process Server Configuration page, set the values for the following parameters.
- **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

15. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
- From the **Select a database product** list, select **Microsoft SQL Server**.
 - Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 157. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.

Table 157. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 158. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.

- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
 19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
 20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
 21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - **default.procctr**: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - **default.procctr.adv**: for a Process Center stand-alone profile for IBM BPM Advanced.
 - **default.procsvr**: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - **default.procsvr.adv**: for a Process Server stand-alone profile for IBM BPM Advanced.
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response

- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPASSWORD perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a stand-alone profile that includes the CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space components in one database.

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates/BPM/
default.procsvr.adv -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath
path to JDBC driver -dbUserId userID -dbPassword password -dbCommonForME true -dbCeiMeSchemaName
ceiSchema -dbSysMeSchemaName sysSchema
-dbAppMeSchemaName appSchema
```

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
```

```

# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#   shortHostNameCellCellNumber

```

```

#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:

```

```

# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.

```

```

# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:

```

```

# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:

```



```

# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
# Database name for Process Server database. The value specified for
# -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrUserId
#
# Description:
# The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrPassword
#
# Description:
# The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName must be different.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter:  dbCommonUserId
#
# Description:
#    The user id to authenticate with the Common database. For Oracle, the
#    default user name is the first three characters of the Oracle database
#    service name. For example: If the database service name is ORCL,
#    dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#    Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#    profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#    and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#    The password for the user id of the dbCommonUserId parameter. Required for
#    all stand-alone IBM Business Process Manager profiles and advanced
#    deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#    to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter:  dbAppMeSchemaName
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbAppMeUserId
#
# Description:
#    User name for application messaging engine.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbAppMePassword
#
# Description:

```

```

# The password for the user id of the dbAppMeUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd
#####
# Parameter: dbCeiMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd
#####
# Parameter: dbSysMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeSchemaName=sys_me_schemaName
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:

```

```

# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false
#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
configureBPC=false

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
```



```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is

```

```

#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:

```

```

# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# MSSQLSERVER_MICROSOFT
# Default Values:
# MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database.  The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password.  Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse.  The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name.  Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```



```

#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
# Parameter: dbCommonUserId
#

```

```

# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####

```

```

# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own

```

```

# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces

```

```

# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNameewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01
#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard
#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.

```

```

# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
processCenterURL=process_Center_URL

```

```

processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####

```

```

# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the

```



```

# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is

```

```

#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbldbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#####
#bpmdbDesign=design_file

#####
# Parameter:  dbCreateNew
#

```

```

# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter:  dbName
#

```

```

# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter:  dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrMePassword
#
# Description:

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database. For Oracle, the
#   default user name is the first three characters of the Oracle database
#   service name. For example: If the database service name is ORCL,
#   dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#   Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#   profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#   and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter. Required for
#   all stand-alone IBM Business Process Manager profiles and advanced
#   deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#   to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid

```

```

dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeSchemaName
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbSysMeSchemaName=sys_me_schemaName
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####

```



```

# Parameter:  environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
#####
# Parameter:  webFormConfig

```

```

#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false
#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration

```

```

# is created.
# The following conditions apply:
# Only DB2 databases are supported.
# The default is set to true for DB2.
# It uses its own schema within the common database (CMNDB) and the schema
# name is always BPEDB.
# Business Process Choreographer database parameters set in the Database
# Design Tool and generated to the database design file are ignored by
# Business Process Choreographer at profile creation time.
# See -bpmdbDesign for information about the database design file.
# Note: If these conditions are too restrictive, consider setting
# -configureBPC to false and using other documented methods to configure
# Business Process Choreographer.
#
# Valid Values:
# false
# Default Values:
# false
#####
configureBPC=false

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.

```

```

#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard

```

```

# production
# Default Values:
# None
#####
#applyPerfTuningSetting=standard
#####
# Parameter: cellName
#
# Description:
# Specifies the cell name of the profile. Use a unique cell name for each
# profile.
# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: serverName
#
# Description:
# Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter: processCenterURL
#
# Description:
# Specifies the URL of the Process Center that the Process Server connects
# to. If this parameter is set, the protocol must be either http or https.

```

```

# Note: No corresponding useServerOffline parameter exists to use the server
# offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterUserId
#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the

```

```

# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default

```

```

# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#

```



```

# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#

```

```

# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter:  dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT

```

```

#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:

```

```

# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
# Database name for Process Server database. The value specified for
# -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrUserId
#
# Description:
# The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrPassword
#
# Description:
# The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
# Database name for Performance Data Warehouse. The value specified for
# -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#

```

```

# Description:
# Performance Data Warehouse user name. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId

```

dbPerfDWMePassword=db_PerfDW_ME_Password

#####

Parameter: dbCommonUserId

#

Description:

The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.

#

Note: Do not use this parameter if you are using -bpmdbDesign.

#

Valid Values:

Varies

Default Values:

None

#

Parameter: dbCommonPassword

#

Description:

The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.

#

Note: Do not use this parameter if you are using -bpmdbDesign.

#

Valid Values:

Varies

Default Values:

None

#

dbCommonUserId=common_db_userid

dbCommonPassword=common_db_pswd

#####

Parameter: dbWinAuth

#

Description:

An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.

#

Note: Do not use this parameter if you are using -bpmdbDesign.

#

Valid Values:

false

true

Default Values:

None

#

#dbWinAuth=true

#####

Parameter: environmentName

#

Description:

Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.

#

Valid Values:

Varies

Default Values:

```

# None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with an SQL Server database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
8. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 159. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	

Table 159. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 160. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with an SQL Server database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 161. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Do not select the Apply Windows Authentication option.	

Table 161. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Common database	<p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database.
Process Server database	<p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database.
Performance Data Warehouse database	<p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer

Table 162. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action required for SQL Authentication	Action required for Windows Authentication
<ul style="list-style-type: none"> • CEI messaging engine schema name • SCA Application bus messaging engine schema name • SCA system bus messaging engine schema name • Process Server messaging engine schema name • Performance Data Warehouse messaging engine schema name 	<p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p>Note: For SQL Authentication, the schema names are mapped to the user IDs.</p>	<p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p>

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

Important: You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Augment** to augment the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - `default.procctr`: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - `default.procctr.adv`: for a Process Center stand-alone profile for IBM BPM Advanced.
 - `default.procsvr`: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - `default.procsvr.adv`: for a Process Server stand-alone profile for IBM BPM Advanced.
 - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response

- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_StandAlone.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring SQL Server databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/ProcessServer/SQLServer/* at profile creation time.
2. Run the following script.

```
configProcessServerDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root/dbscripts/PerformanceDW/SQLServer/* at profile creation time.
2. Run the following script.

```
configPerformanceDWDB.sh
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the Common database during the configuration.
- Created the database that you want to create the Common database tables in.

1. Go to the directory containing the **configCommonDB.sh** script. The default directory is *install_root/profile/dbscripts/CommonDB/SQLServer/*.
2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

configCommonDB.sh createDB - creates the database and tables

configCommonDB.sh - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.sh** script. The default directory is *install_root/profile/dbscripts/BusinessSpace/node_name_server_name/SQLServer/CMNDB*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:
configBusinessSpaceDB.sh Alternatively, you could customize and run the individual scripts manually in the following order:
 - a. **createDatabase.sql**
 - b. **createTable_BusinessSpace.sql**
 - c. **createGrant_BusinessSpace.sql**

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Database passwords must be encrypted.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root/profiles/ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.sh**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Important: If the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
```



```
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Modifying an existing installation:

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

Configuring a network deployment environment on Solaris

For a network deployment configuration, you install the software and then create or augment a deployment manager, create or augment at least one custom profile, and configure the network deployment environment.

Granting write permission of files and directories to nonroot users for profile creation or augmentation:

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be a root or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonroot users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root/profiles* directory of the product.

Restrictions:

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. A nonroot user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a root or nonroot user.
2. Using operating system commands, perform the following steps:
 - a. Create a group named **profilers**, which will contain all users who can create profiles.
 - b. Create a user named **user1**, who can create profiles.
 - c. Add users **product_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
 - Create the *install_root/logs/manageprofiles* directory:
`mkdir install_root/logs/manageprofiles`
 - Create the *install_root/properties/fsdb* directory:
`mkdir install_root/properties/fsdb`
5. If no profile exists, create the *profileRegistry.xml* file as the installer. For this example, the file path is:
`install_root/properties/profileRegistry.xml`

Add the following information to the *profileRegistry.xml* file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **/opt/IBM/WebSphere/AppServer**.

```
export WASHOME=/opt/IBM/WebSphere/AppServer
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
install_root/properties/profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonroot user to create profiles.

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

The root user and the nonroot user can use the same tasks to manage profiles.

Creating databases and database design files:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. You must create the databases before network deployment.

Creating databases and database design files for DB2:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

Creating DB2 databases:

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the /etc/profile file. Run

```
vi /etc/profile
```

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root user to the /etc/group administrative groups. Run

```
vi /etc/group
```

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102:root
db2fadm: |:103:db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
- When the DB2 database manager is not running:
SQL1032N No start database manager command was issued. SQLSTATE=57019
- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:
SQL1092N "ROOT" does not have the authority to perform the requested command.

Creating database design files for DB2:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```

`-v db_design_file | db_scripts_output_directory`
when a `db_design_file` is given, validation will be done on the specified database design file based on the database specs.
When a `db_scripts_output_directory` is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

`-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]`
generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use `BSPACE` as the database name. If you need the database name to be `BSPACE`, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the `[begin] = BSpace : WBI_BSPACE` section, changing the `databaseName` value to `BSPACE`.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated `BSPACE` for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to `BSPACE` in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the `DbDesignGenerator.sh` command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5)wesb.nd.topology
- (6)wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI_BPC : [status = not complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = not complete]
- (6)[SibME] WBI_BPC_ME : [status = not complete]
- (7)[SibME] WBI_CEI_ME : [status = not complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = not complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

- (1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components,  
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]  
(2)[BPC] WBI_BPC : [status = complete]  
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]  
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]  
(5)[BSPACE] WBI_BSPACE : [status = complete]  
(6)[SibME] WBI_BPC_ME : [status = complete]  
(7)[SibME] WBI_CEI_ME : [status = complete]  
(8)[SibME] WBI_SCA_APP_ME : [status = complete]  
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]  
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]  
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]  
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the master component, all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]  
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]  
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]  
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]  
(5)[save and exit]
```

```
Please enter the number for the database component :5
```

```
[status] bpm.standard.nd is complete with 0 remaining item(s):
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm_performancedw
- (4)bpm_processserver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=50000] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
WAS_HOME/profiles/default/util/dbUtils/DB2-distributed-CommonDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating database design files and database scripts (Solaris network deployment):

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files and database scripts for deployment environments with DB2 for z/OS:

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the `[begin] = BSpace : WBI_BSPACE` section, changing the `databaseName` value to `BSPACE`.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the `DbDesignGenerator.sh` command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and

press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, `WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB`. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
```


Database schema name (SQLID) [default=] :
Bufferpool of 4K size for tables [default=BP1] :
Bufferpool for indexes [default=BP2] :
Bufferpool of 4K size for LOB data [default=BP3] :
Bufferpool of 8K size for tables [default=BP8K1] :
Bufferpool of 16K size for tables [default=BP16K1] :
Bufferpool of 32K size for tables [default=BP32K1] :
Storage group name [default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS  
(2)DB2 Universal JDBC Driver Provider on local z/OS
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database Location on z/OS system [default=] :  
Database server host [default=] :  
Database server port [default=] :  
Data source user name [default=] :  
Data source password [default=] :  
DB2 Universal JDBC driver path [default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
Universal JDBC driver path [default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
DB2 Universal JDBC driver native path (This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4) [default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.
```

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibME] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibME] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

Note: For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1)false
- (2>true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.nd.topology**, after you configure the master component, you must manually configure the **BPM_PerformanceDW** and **BPM_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :12

```
[status] bpm.advanced.nd.topology is not complete with 6 remaining item(s):
[ 1 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[ 2 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'catalog' for VCAT is empty.
[ 3 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminUserName' for SECURITY_ADMIN_USER is empty.
[ 4 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
[ 5 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[ 6 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'catalog' for VCAT is empty.
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment:

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

- ? , -help
display help info.
- e db_design_file_name
edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
when a db_design_file is given, validation will be done on the specified database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment  
(2)Create a database design for a single component  
(3)Edit an existing database design  
(4)Generate database scripts from a database design  
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc  
(2)bpcreporting  
(3)bpm_performancedw  
(4)bpm_processserver  
(5)bspace  
(6)cei  
(7)sca  
(8)sibme  
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. To generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Related tasks:

“Creating the messaging engine data stores” on page 2497

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the

messaging engine data stores.

Generating scripts for creating database objects in DB2 for z/OS:

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
 - Database server details
 - Data source details
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and

generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory /tmp/DB2-zOS/W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as /tmp/DB2-zOS/W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as /tmp/DB2-zOS/W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd /install_root/util/dbUtils
```

For example: cd /opt/IBM/WebSphere/AppServer/util/dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.sh -g db_design_file
```

where *db_design_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.sh -g /tmp/DB2-zOS/W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

Tip: In the database design file, the directories for the SQL scripts are defined by the *ddl_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

Note: The `createDB2.sh` script requires execute permissions.

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required `userName` and `password` properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Setting up databases for Oracle:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a `BPMDBA` user name for the Process Server database, a `PDWDBA` user name for the Performance Data Warehouse database, and a `CMNDBA` user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```


4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;  
SQL> GRANT execute on DBMS_LOCK to PDWDBA;  
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating database design files for Oracle:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with Oracle:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help  
    display help info.
```

```
-e db_design_file_name  
    edit the specified database design file (e.g. *.dbDesign, *.properties).
```

```
-v db_design_file | db_scripts_output_directory  
    when a db_design_file is given, validation will be done on the specified  
    database design file based on the database specs.
```

```
When a db_scripts_output_directory is given, the database scripts  
in the specified directory will be validated. Currently only  
scripts generated from template ddl generator can be validated.
```

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
```

- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI_BPC : [status = not complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = not complete]
- (6)[SibME] WBI_BPC_ME : [status = not complete]
- (7)[SibME] WBI_CEI_ME : [status = not complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = not complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict.

If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components,  
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]  
(2)[BPC] WBI_BPC : [status = complete]  
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]  
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]  
(5)[BSPACE] WBI_BSPACE : [status = complete]  
(6)[SibME] WBI_BPC_ME : [status = complete]  
(7)[SibME] WBI_CEI_ME : [status = complete]  
(8)[SibME] WBI_SCA_APP_ME : [status = complete]  
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]  
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]  
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]  
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.nd.topology**, you must manually configure the **BPM_PerformanceDW**, **BPM_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]  
(2)[BPC] WBI_BPC : [status = complete]  
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]  
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]  
(5)[BSPACE] WBI_BSPACE : [status = complete]  
(6)[SibME] WBI_BPC_ME : [status = complete]  
(7)[SibME] WBI_CEI_ME : [status = complete]  
(8)[SibME] WBI_SCA_APP_ME : [status = complete]  
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]  
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]  
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]  
(12)[save and exit]
```

```
Please enter the number for the database component :3
```

```
-----  
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminUserName' for SECURITY_ADMIN_USER is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects :
required property 'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named dbDesignGenerator.log is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
```

When a `db_scripts_output_directory` is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...  
[db_design_fileN] [-d output_directoryN]
```

generate the database scripts from the specified design files in batch mode. The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment  
(2)Create a database design for a single component  
(3)Edit an existing database design  
(4)Generate database scripts from a database design  
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc  
(2)bpcreporting  
(3)bpm_performancedw  
(4)bpm_processserver  
(5)bspace  
(6)cei  
(7)sca  
(8)sibme  
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=CMNDB] :

Database User name[default=] :

Database schema [default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

Database Location(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:
- generate database scripts? (y/n) [default=y] :
10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/util/dbUtils/Oracle-CommonDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating databases and database design files for SQL Server:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

Tip: If you are using the federated repositories as a user registry, you can ignore warnings in the `systemout.log` file about maximum key length: ... **Warning! The maximum key length is 900 bytes ...** If you are using the stand-alone LDAP registry, ensure that the number of characters in all the user distinguished name (DN) entries in your organization do not exceed the 131 character limit. If the number of characters in any of the user DN entries exceeds 131 characters, you must change the user account registry to the federated repositories option.

Creating SQL Server databases:

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password  
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name  
'message.routingCache'  
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is  
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class  
[com.lombardiisofware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;  
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;  
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is  
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.  
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement  
(SQLServerPreparedStatement.java:388)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute  
(SQLServerPreparedStatement.java:338)  
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)  
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)  
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery  
(SQLServerPreparedStatement.java:281)  
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)  
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)  
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)  
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.  
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)  
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmntExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.

- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 163.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.

- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating database design files for SQL Server:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for deployment environments with SQL Server:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root/Util/dbUtils* directory. For example:

```
/opt/IBM/WebSphere/AppServer/Util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
```

help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1) bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.

[info] Completed database components can be edited to change existing or defaulted property values.

[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI_BPC : [status = not complete]
- (3) [BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (4) [BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI_BSPACE : [status = not complete]
- (6) [SibME] WBI_BPC_ME : [status = not complete]
- (7) [SibME] WBI_CEI_ME : [status = not complete]
- (8) [SibME] WBI_SCA_APP_ME : [status = not complete]
- (9) [SibME] WBI_SCA_SYS_ME : [status = not complete]
- (10) [SibMe] BPM_PerformanceDW_ME : [status = not complete]
- (11) [SibMe] BPM_ProcessServer_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1) [WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI_CommonDB is not complete with 1 remaining item(s):

[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI_CommonDB is complete with 0 remaining item(s):

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
- (2)[BPC] WBI_BPC : [status = complete]
- (3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
- (4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
- (5)[BSpace] WBI_BSPACE : [status = complete]
- (6)[SibME] WBI_BPC_ME : [status = complete]
- (7)[SibME] WBI_CEI_ME : [status = complete]
- (8)[SibME] WBI_SCA_APP_ME : [status = complete]
- (9)[SibME] WBI_SCA_SYS_ME : [status = complete]
- (10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the **BPM_ProcessServer**, you must manually configure the **BPM_PerformanceDW** component for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
- (2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
- (3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
- (4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
- (5)[save and exit]

Please enter the number for the database component :2

```
-----  
[status] BPM_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB_USER is empty.
```

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with SQL Server in a network deployment environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the `DbDesignGenerator.sh` command from the `install_root/util/dbUtils` directory. For example:

```
/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commond
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :

Data source password[default=] :

MS SqlServer JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=/WebSphere/V8T8DM/DeploymentManager/util/dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in  
WAS_HOME/profiles/default/util/dbUtils/SQLServer-CommonDB for WBI_CommonDB  
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Configuring the network deployment environment using the `configureNode` command:

To configure a typical network deployment environment, you can use the `configureNode` command instead of using the Profile Management Tool and Deployment Environment wizard or the `manageprofiles` and `wsadmin` commands. This command provides limited customization by using a properties file, and creates a ready-to-use environment. You can also use `configureNode` later to extend the deployment environment.

Restriction: This command is not supported if your database type is DB2 for z/OS.

Important: The `configureNode` command cannot be run twice on one machine without first deleting the profile that was created. To run the `configureNode` command a second time, first delete the profile.

Creating the deployment environment using the `configureNode` command:

You can use the `configureNode` command to create a typical network deployment environment. The goal of this command is to provide a ready-to-use environment. The only parameters that can be customized are provided in a properties file.

Restriction: This command is not supported if your database type is DB2 for z/OS.

You must have installed the product and created all the databases that are specified in the properties file.

Important: You must have already installed the database software and created the databases (but do not create tables). You must also have created all the users that you specify in the properties file. Make sure the users have the installation privileges listed in Database privileges.

For your SQL Server database server:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
- If connections to the database will be made by the current Windows user that the server is running under, the SQL Server must have **Windows authentication mode** or **SQL Server and Windows Authentication mode** enabled, as specified through Microsoft SQL Server Management Studio.

Restriction: Run `configureNode` commands sequentially; do not try to run the command on multiple machines at the same time.

The `configureNode` command uses default ports to create the deployment manager. You cannot customize the ports using this command. The parameters listed in the sample files are the only parameters that can be customized. Make sure that no other profiles that are configured with default ports are started on the same machine.

The `configureNode` command creates the following network deployment topology:

- A single cell environment
- The environment contains the number of cluster members that you specify in the properties file
- The environment uses the Remote Messaging, Remote Support, and Web topology pattern, which includes the following four clusters:
 - Messaging infrastructure cluster

- Supporting infrastructure cluster
- Application deployment target cluster
- Web application cluster

The command performs the following tasks:

- Creates the deployment manager node based on the values in the deployment manager properties file and starts the deployment manager.
- Creates the deployment environment definition.
- Creates a custom node based on the values in the custom node properties file.
- Federates the node and adds the node to the deployment environment.
- Generates the deployment environment.
- Creates the database tables. (You must already have created the databases. See "Before you begin" for instructions.)
- Runs the bootstrap utility to load the Process Server database with system information.
- Configures Business Space and Process Portal on the web application cluster.

The default port is used to create the deployment manager if the optional ports are not specified in the properties file. If a deployment manager is already running on the machine when you run **configureNode**, creating the deployment environment fails. If a deployment manager has already been created but is not running, you must change the port of one of the deployment managers or you might get a port conflict error when you start the other deployment manager.

To create the deployment environment for the first time, complete the following steps:

1. On the machine where you want to create the deployment environment, locate the sample properties file: *install_root/util/ndUtils/samples/*.
2. Copy the sample files and modify the files to reflect your environment. For example, for a deployment manager for IBM BPM Advanced for Process Server, choose the *sample_adv_pc_dmgr.properties* file.

For each database that you want to create, copy the database parameters section and specify the database name, user name, password, and schema name. The Process Server database, Performance Data Warehouse database, and Common database are required, and they must have different names. For IBM BPM Advanced, the Business Process Choreographer database is also required.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

Restriction: The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the names of the two files you created. The custom node file is optional, but if you do not include it, only the deployment manager is created. For example:

```
install_root/util/ndUtils/configureNode.sh -dmgr_response my_dmgr_response_file.properties  
-response node01_response_file.properties
```

Note: If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to **127.0.0.1 localhost** in the hosts file. For example: **127.0.0.1 localhost example.ibm.com**.

You now have a deployment manager, a custom node, and a deployment environment that use the Remote Messaging, Remote Support, and Web topology pattern.

Messages are recorded in the file *install_root/logs/config/configureNode.log*.

Tip: After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception

`com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT.` This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

 Virtual hosts

Related tasks:

 Configuring virtual hosts

Related reference:

 Test connection service

Extending the deployment environment using the `configureNode` command:

After you have created a deployment environment using the **configureNode** command, you can run the command at a later time to extend the deployment environment.

You need the following prerequisites before running the **configureNode** command:

- You must have installed the product on the machine where you want to extend the deployment environment.
- The deployment manager must be running.
- You must have created the deployment environment by running the **configureNode** command.

Restriction: Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.

1. On the machine where you want to extend the deployment environment, locate the sample custom node files: `install_root/util/ndUtils/samples/`
2. Copy the sample file that is most similar to the properties file that you need. For example, to extend the deployment environment for IBM BPM Advanced for Process Server, choose the `sample_adv_ps_node.properties` file.

Modify the new custom node file to reflect your environment, including the details of the deployment manager.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

Restriction: The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the name of the sample file. For example:

`install_root/util/ndUtils/configureNode.sh -response node01_response_file.properties`

Note: If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to `127.0.0.1 localhost` in the hosts file. For example: `127.0.0.1 localhost example.ibm.com`.

You have extended the deployment environment.

Messages are recorded in the file `install_root/logs/config/configureNode.log`.

Tip: After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception `com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT`. This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

 [Virtual hosts](#)

Related tasks:

 [Configuring virtual hosts](#)

Related reference:

 [Test connection service](#)

Configuring profiles and creating a network deployment environment:

After you install the product, you must create or augment a deployment manager and one or more custom profiles to define the runtime environment. Before starting the deployment manager, you must have configured the databases that are to be used with IBM Business Process Manager.

Creating or augmenting network deployment profiles:

If you used the **configureNode** command to create and configure your deployment environment, no further configuration is required. Otherwise, you must create or augment a deployment manager profile and one or more custom profiles before creating the deployment environment. Using profiles, you can have more than one runtime environment on a system, without having to install multiple copies of IBM Business Process Manager.

Creating or augmenting network deployment profiles with a DB2 database server:

You can configure a network deployment environment for IBM Business Process Manager using a DB2 database server.

Preparing the DB2 database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components. You can either create a new database and tables or postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

If you plan to configure the Common database on a remote server, you must have created the database before beginning to create or augment the profile. You can create a database on the local server or use an existing one on a remote server.

If you plan to use DB2 Universal Database, as part of preparing your database server:

- You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands during profile creation. Add the **db2profile** script to the `/etc/profile` file. Run
`vi /etc/profile`

and add the following lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the `db2profile` script to the `/etc/profile` directory, you must run the **db2profile** script to set the DB2 environment.

- You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using `db2inst1` as the user ID, add the root user to the `/etc/group` administrative groups. Run
`vi /etc/group`

and update the following lines:

```
dasadm: |:101:dasusr1,db2inst1,root
db2iadm: |:102;root
db2fadm: |:103;db2fenc1,root
```

If the procedure is not followed, you might see the following exceptions when creating or augmenting a profile.

- When the **db2profile** script is not run:

```
/opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

- When the DB2 database manager is not running:

```
SQL1032N No start database manager command was issued. SQLSTATE=57019
```

- When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

```
SQL1092N "ROOT" does not have the authority to perform the requested command.
```

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the `install_root/.Xdefaults` file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
- Run the command *install_root/bin/ProfileManagement/pmt.sh*.

2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.

5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.

6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.

7. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Advanced:** Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.

- c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 164. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with a DB2 database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.
11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.
If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The

password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - e. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 165. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdriivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create a deployment manager profile.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response

- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```


If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMgr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
```

create

```
#####  
# Parameter: templatePath  
#  
# Description:  
# Specifies the directory path to the template files in the installation  
# root directory. Within the profileTemplates directory are various  
# directories that correspond to different profile types and that vary with  
# the type of product installed. The profile directories are the paths that  
# you indicate while using the -templatePath option.  
# Use absolute paths. This parameter must exist as a directory and point to  
# a valid template directory.  
#  
# When using the -templatePath parameter, specify the fully qualified file  
# path for the parameter.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
templatePath=BPM/dmgr.procctr.adv
```

```
#####  
# Parameter: profileName  
#  
# Description:  
# Specifies the name of the profile. Use a unique value when creating a  
# profile.  
# Each profile that shares the same set of product binaries must have a  
# unique name. The default profile name is based on the profile type and a  
# trailing number, for example:  
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,  
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a  
# unique profile name. The value for this parameter must not contain spaces  
# or characters that are not valid such as the following: *, ?, ", <, >, ,,  
# /, \, and |. The profile name that you choose must not be in use. On  
# Windows platforms: If the fully qualified path contains spaces, enclose  
# the value in quotation marks. The default value is based on the  
# install_root directory, the profiles subdirectory, and the name of the  
# file. For example, the default for profile creation is:  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties  
# file in the install_root/properties directory. The value for this  
# parameter must be a valid path for the target system and must not be  
# currently in use. You must have permissions to write to the directory.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
profileName=Dmgr01
```

```
#####  
# Parameter: cellName  
#  
# Description:  
# Specifies the cell name of the profile. Use a unique cell name for each  
# profile.  
# The default value for this parameter is based on a combination of the  
# short host name, the constant Cell, and a trailing number, for example:  
# if (DMgr)  
# shortHostNameCellCellNumber  
# else  
# shortHostNameNodeNodeNumberCell
```



```

# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The

```

```

#    localsystem value runs the Windows service under the local account of the
#    user who creates the profile. The default value for this parameter is
#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic

```

```

# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbidbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath

```

```

#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter:  dbName
#
# Description:

```

```

# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test

```

```

# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .

```



```

#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#

```

```

# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#

```

```

# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile. Do not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port

```

```

# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:

```

```

# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false

```

```

# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.

```

```

#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:

```



```

# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the

```

```

# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:

```

```

# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####

```

```

# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#

```

```

# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter:  dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:

```

```

# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a

```

```

# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath

```



```

#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#

```

```

# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####

```

```

#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with a DB2 database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
 11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.

- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 166. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles for Process Server with a DB2 database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.

- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **DB2**.

- b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 167. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:
 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response

- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters.
For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.

- Run the command `install_root/bin/ProfileManagement/pmt.sh`.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
 9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled

- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.

- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced**: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the `manageprofiles` command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create custom profiles.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the `-templatePath` option).

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.

- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response

- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
```



```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber

```

```

#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  enableService
#
# Description:
#    Enables the creation of a Linux service. Valid values include true or
#    false. The default value for this parameter is false. When the
#    manageprofiles command-line utility is run with the -enableService option
#    set to true , the Linux service is created with the profile when the
#    command is run by the root user. When a nonroot user runs the
#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#    Specifies the user ID that is used during the creation of the Linux
#    service so that the Linux service will run under this user ID. The Linux
#    service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#    The type of the owner account of the Windows service created for the
#    profile. Valid values include specifieduser or localsystem. The
#    localsystem value runs the Windows service under the local account of the
#    user who creates the profile. The default value for this parameter is
#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system

```

```

# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.

```

```

#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is

```

```

# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test

```

```
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
```

```

# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:

```



```

#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root

```

```

# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType

```

```

#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort

```

```

# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####

```

```

# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####

```

create

```
#####  
# Parameter: templatePath  
#  
# Description:  
# Specifies the directory path to the template files in the installation  
# root directory. Within the profileTemplates directory are various  
# directories that correspond to different profile types and that vary with  
# the type of product installed. The profile directories are the paths that  
# you indicate while using the -templatePath option.  
# Use absolute paths. This parameter must exist as a directory and point to  
# a valid template directory.  
#  
# When using the -templatePath parameter, specify the fully qualified file  
# path for the parameter.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
templatePath=BPM/managed.procsvr.adv
```

```
#####  
# Parameter: profileName  
#  
# Description:  
# Specifies the name of the profile. Use a unique value when creating a  
# profile.  
# Each profile that shares the same set of product binaries must have a  
# unique name. The default profile name is based on the profile type and a  
# trailing number, for example:  
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,  
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a  
# unique profile name. The value for this parameter must not contain spaces  
# or characters that are not valid such as the following: *, ?, ", <, >, ,,  
# /, \, and |. The profile name that you choose must not be in use. On  
# Windows platforms: If the fully qualified path contains spaces, enclose  
# the value in quotation marks. The default value is based on the  
# install_root directory, the profiles subdirectory, and the name of the  
# file. For example, the default for profile creation is:  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties  
# file in the install_root/properties directory. The value for this  
# parameter must be a valid path for the target system and must not be  
# currently in use. You must have permissions to write to the directory.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
profileName=Custom01
```

```
#####  
# Parameter: cellName  
#  
# Description:  
# Specifies the cell name of the profile. Use a unique cell name for each  
# profile.  
# The default value for this parameter is based on a combination of the  
# short host name, the constant Cell, and a trailing number, for example:  
# if (DMgr)  
# shortHostNameCellCellNumber  
# else  
# shortHostNameNodeNodeNumberCell
```

```

# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:

```



```

#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual

```

```

#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  dbType
#
# Description:
#    The database type. Set one of the following values for the type of
#    database product you are using with IBM Business Process Manager.
#
#    Required only for one of the following:
#    A stand-alone profile when -bpmdbDesign was not specified.
#    An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2_DATASERVER
# Default Values:
#    DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#    The directory path location of JDBC driver files.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
#####
# Parameter:  federateLaterBPM
#
# Description:
#    Indicates if the managed profile is to be federated later using either the
#    profileTemplates/managed.esbserver or the
#    profileTemplates/BPM/managed.templateName template. Valid values are true
#    or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#    -dmgrAdminPassword parameters are not set, the default value for this
#    parameter is true. If you use -federateLaterBPM, you must either omit the
#    WebSphere Application Server-based parameter -federateLater or set it to
#    true.
#
# Valid Values:
#    false

```

```

# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dmgrPort
#
# Description:
# Identifies the SOAP port of the deployment manager. Specify this parameter
# and the dmgrHost parameter to federate a custom profile as it is created
# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#

```

```

# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal

```

```

# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On

```

```

# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####

```

```

# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or

```



```

# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword

```

```

#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#

```

```

# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a

```

```

# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-*10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command **install_root/bin/ProfileManagement/pmt.sh**.
 3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
- `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.

- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response

- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:


```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):


```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

Creating or augmenting network deployment profiles with a DB2 for z/OS database server:

You can configure a network deployment environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.

- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
14. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
15. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool on Solaris:

You can configure a deployment manager profile for Process Server using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
14. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
15. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:
 - `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.
 - `dmgr.procctr.adv`: for a Process Center deployment manager profile for IBM BPM Advanced.
 - `dmgr.procsvr`: for a Process Server deployment manager profile for IBM BPM Standard.
 - `dmgr.procsvr.adv`: for a Process Server deployment manager profile for IBM BPM Advanced.
 - `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2zOS.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2zOS.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2zOS.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2zOS.response`
- `PC_Adv_Managed_Oracle.response`

- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters.
For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
```

```

#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the

```

```

# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the

```

```

# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the

```

```

# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```



```

# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled

```

```

# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2UDBOS390

```

```

# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB

```

dbUserId=db_userid
dbPassword=db_pswd

#####

Parameter: dbStorageGroup

#

Description:

The storage group name for DB2 z/OS databases.

#

Note: Do not use this parameter if you are using -bpmdbDesign.

#

Valid Values:

Varies

Default Values:

None

#

Parameter: dbConnectionLocation

#

Description:

The location of DB2 for z/OS database.

#

Note: Do not use this parameter if you are using -bpmdbDesign.

#

Valid Values:

Varies

Default Values:

None

#####

dbStorageGroup=db_stg_group

dbConnectionLocation=db2_location

#####

Parameter: environmentType

#

Description:

Specifies the environment type. The environment type refers to how IBM Business Process Manager is used; for example, in a production, stage or test environment. Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify Stage as the Environment type if the server you are configuring will be accessed and used to review content and new functionality. Valid values are as follows:

Test

Use Test if the server you are configuring is to be used as a testing environment.

Stage

Use Stage if the server is to serve as a staging platform to be used as a preproduction server.

Production

Use Production if the server is to serve in a production capacity.

The default value is Test.

#

Valid Values:

Test

Production

Stage

Default Values:

Test

#####

environmentType=Test

#####

Parameter: isDeveloperServer

#

Description:

Specifies whether the server is intended for development purposes only.

```

# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####

```

```

# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters

```

```

# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None

```

```

#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```



```

#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.

```

```

#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual

```

```

#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
```

```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber

```

```

# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

```

```

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#

```



```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#    Indicates a new database is created or if an existing database is to be
#    reused. Valid values are true or false. This value cannot be true for an
#    Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#    a DB2 database.
#
# Valid Values:
#    false
# Default Values:
#    false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#    The database type. Set one of the following values for the type of
#    database product you are using with IBM Business Process Manager.
#
#    Required only for one of the following:
#    A stand-alone profile when -bpmdbDesign was not specified.
#    An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2UDBOS390
# Default Values:
#    DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: dbDelayConfig
#
# Description:
#    Indicates whether to postpone table creation until after the profile is
#    created. Valid values are true or false. The parameter is set to false by
#    default. Setting this parameter to true delays the execution of the
#    scripts that create the database tables. No. Specify false if you want the
#    database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#####
dbDelayConfig=true

#####
# Parameter: dbName
#
# Description:
#    The name of the database. By default, the value is set to orcl for Oracle
#    databases and to CMNDB for all other supported databases.
#

```

```

# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: dbStorageGroup
#
# Description:
#   The storage group name for DB2 z/OS databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbConnectionLocation
#
# Description:
#   The location of DB2 for z/OS database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

```

```
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
```

```

# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:

```

```

# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies

```

```

#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option

```



```

# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:

```

```

# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.

```

```

#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 8. **Advanced:** On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 9. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other

WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

14. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 8. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain

keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

14. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
 - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.

15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Run the scripts to create and configure the DB2 for z/OS database.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response

- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default

language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.

15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response

- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
```

```

# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters

```

```

# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#
# Parameter: nodeName
#
# Description:
# Specifies the node name for the node that is created with the new profile.
# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
# Varies
# Default Values:
# Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None

```

```

#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType

```

```

#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.

```

```

#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2UDBOS390
# Default Values:
#   DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or

```

```

# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production

```



```

# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT
#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2zOS.response
#

```

```

# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

```

```

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

```

```

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true

```

```

# Default Values:
#   None
#
# Parameter:  serviceName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this

```

```

#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.  If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the

```

```

# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:

```

```

# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses DB2z0S.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2z0S.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .

```



```

#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#

```

```

# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#

```

```

# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName

```

```

#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false

```

```

# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2UDBOS390
# Default Values:
# DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#

```

```

# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a

```

```

# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****

```

```

# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:

```



```

# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative

```

```

# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the

```

```

# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.

```

```

# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values

```

```

# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```

Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1

```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
 3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:
 - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
 - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
 - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter.

The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response

- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:

```
profile_root/bin/startManager.sh
```

- The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 Add managed node settings (WebSphere Application Server)

Creating and configuring DB2 for z/OS databases after network deployment profile creation:

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

DB2 for z/OS authorization prerequisites:

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

User authorization requirements for DB2 for z/OS

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as *#DB_USER#*. These permissions are available in the *createDatabase.sql* database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the #DB_USER# user to create sequences and stored procedures with a schema qualifier of #SCHEMA#:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

Authorization requirements for views on DB2 for z/OS V10

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database #DB_NAME# can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE SICELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
```



```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;  
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;
```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named S1CELL, and connect user WSADMIN to it, as follows:

```
INFORMATION FOR GROUP S1CELL  
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144  
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK  
NO MODEL DATA SET  
TERMUACC  
NO SUBGROUPS  
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=  
WSADMIN CONNECT 000000 NONE  
CONNECT ATTRIBUTES=NONE  
REVOKE DATE=NONE RESUME DATE=NONE
```

Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating databases in the DB2 for z/OS subsystem:

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

Considerations for choosing the createDB2.sh script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).

- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two .sql files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

Related tasks:

“Creating database design files and database scripts (Solaris network deployment)” on page 2260
 You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Configuring the DB2 command line processor:

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
 - Modify the CLASSPATH environment variable to include the `clp.jar` file.
 - Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
 - Define the **db2** command as an alias for the command that starts the command line processor.
 - Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:

- A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
- A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

Tip: When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script”

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2493

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (*z_schema.sql* and *z_schemaProc.sql*) that you can run later by using the DB2 command line processor.

Creating DB2 for z/OS database objects using the createDB2.sh script:

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTPE2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database

configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCreate  
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

-RunSQL

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

Tip: The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 2490

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using the DB2 command line processor:

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database

configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.

- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

Important: It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate  
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

-DBAlias

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

-DBName

Specifies the name of the database to be created.

-DBSto

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

-DBCcreate

Creates the database.

-DBVCat

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

-DBUser

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

Tip: By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two .sql files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCCreate -DBVCat DSNV10PP  
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

The z_schema.sql and z_schemaProc.sql files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The z_schemaProc.sql file contains SQL statements for stored procedures, and is generated in addition to z_schema.sql because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the z_schema.sql file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the z_schemaProc.sql file, and additionally use the **-td@** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

Related tasks:

“Configuring the DB2 command line processor” on page 2490

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

Tip: The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

Tip: If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

```
@STOGRP@ = DB2 Storage group name
@DB_NAME@ = DB2 Database name
@SCHEMA@ = DB2 SQLID (Schema Qualifier)
@BP4K@ = Buffer Pool of 4k Size
@BPINDEX@ = Buffer Pool Index
@VOLUMES@ = DB2 comma separated list of volume-ids
@VCAT@ = DB2 integrated catalog facility name
@DB_USER@ = The user ID which will be used to create the database objects
```

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

Tip: If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (z_schema.sql and z_schemaProc.sql), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

Related tasks:

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2493

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z_schema.sql and z_schemaProc.sql) that you can run later by using the DB2

command line processor.

Creating the messaging engine data stores:

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Before you create and run the database scripts:

- Create a cluster.
- Decide which service integration bus components you want to create in your network deployment configuration.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
 - When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
 - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
 - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
 - a. Run the database design tool (**DbDesignGenerator**) from the *install_root/Util/dbUtils* directory.
 - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
 - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

Tip: As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

Related tasks:

“Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment” on page 2267

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for deployment environments with DB2 for z/OS” on page 2260

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

Note: Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

Setting the correct schema name for the messaging engines:

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
 - a. Select **Messaging engines**, and then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Verify that the messaging engines can be started:

- a. Log out of the administrative console.
- b. Stop, and then restart the server.
- c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

Creating or augmenting network deployment profiles with an Oracle database server:

You can configure a network deployment environment for IBM Business Process Manager using an Oracle database server.

Preparing the Oracle database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

To configure the Common database while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 168. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.
11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.
If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The

password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 169. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response

- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
```

```

# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv
#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumber where profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01
#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod

```



```

#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile

```



```

#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:

```

```

# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true

```

```

#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter:  bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter:  dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.

```

```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####

```

```

# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied

```

```

# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a

```

```

# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name

```


nodeName=node_name

```
#####  
# Parameter: enableAdminSecurity  
#  
# Description:  
# For IBM Business Process Manager omit this parameter. Administrative  
# security is always enabled for IBM Business Process Manager profiles.  
# You must also specify the parameters -adminUserName and -adminPassword  
# along with the values for these parameters.  
#  
# Valid Values:  
# true  
# Default Values:  
# true  
#####  
enableAdminSecurity=true  
adminUserName=adminUser_ID  
adminPassword=adminPassword  
  
#####  
# Parameter: signingCertDN  
#  
# Description:  
# Specifies the distinguished name of the root signing certificate that you  
# create when you create the profile. Specify the distinguished name in  
# quotation marks. This default personal certificate is located in the  
# server keystore file. If you do not specifically create or import a root  
# signing certificate, one is created by default. See the  
# -signingCertValidityPeriod parameter and the -keyStorePassword.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#  
# Parameter: signingCertValidityPeriod  
#  
# Description:  
# An optional parameter that specifies the amount of time in years that the  
# root signing certificate is valid. If you do not specify this parameter  
# with the -signingCertDN parameter, the root signing certificate is valid  
# for 20 years.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#  
# Parameter: keyStorePassword  
#  
# Description:  
# Specifies the password to use on all keystore files created during profile  
# creation. Keystore files are created for the default personal certificate  
# and the root signing certificate.  
#  
# Valid Values:  
# Varies  
# Default Values:  
# None  
#####  
#signingCertDN =distinguished_name  
#signingCertValidityPeriod =validity_period  
#keyStorePassword =keyStore_password  
  
#####  
# Parameter: defaultPorts
```

```

#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the

```

```

# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM

```

```

# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,

```

```

# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose

```

```

# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity

```

```

#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.

```



```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbiddbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file
#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.

```

```

#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
# Parameter: dbLocation
#
# Description:
#   The directory of the ORACLE_HOME directory. This parameter is required
#   when the parameter dbDelayConfig is set to false. No, this parameter is
#   only valid if you are using Oracle databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbLocation=db_location

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####

```

```

#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:

```

```

# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMGr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

```

```

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.

```



```

# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is

```

```

#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:

```

```

# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbDriverType=ORACLE_THIN

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when

```

```

#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with an Oracle database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```

Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1

```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.

- Run the command *install_root/bin/ProfileManagement/pmt.sh*.
2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the *bin* directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 170. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles for Process Server with an Oracle database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.
13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.
If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root

signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Oracle**.
 - b. Optional: If you want to change the default values, enter new names for the required databases.
 - The database names can be the same.
 - You must input the database service name for each database.
 - The database must already exist.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 171. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.

- Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the `install_root/properties/profileRegistry.xml` file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:

- `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.
- `dmgr.procctr.adv`: for a Process Center deployment manager profile for IBM BPM Advanced.
- `dmgr.procsvr`: for a Process Server deployment manager profile for IBM BPM Standard.
- `dmgr.procsvr.adv`: for a Process Server deployment manager profile for IBM BPM Advanced.
- `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile.

3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2zOS.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2zOS.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`

- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_Standalone_DB2.response
- PC_Std_Standalone_DB2zOS.response
- PC_Std_Standalone_Oracle.response
- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2zOS.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```


Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
6. If you selected **Typical** profile creation, skip to the Federation step.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root/profiles/profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

- Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

- Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.

- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the `manageprofiles` command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create custom profiles.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the `-templatePath` option).

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
 - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
 - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
 - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
 - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2z0S.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`

- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```
#####  
templatePath=BPM/managed.procctr.adv
```

```
#####
```

```
# Parameter: profileName
```

```
#
```

```
# Description:
```

```
# Specifies the name of the profile. Use a unique value when creating a  
# profile.  
# Each profile that shares the same set of product binaries must have a  
# unique name. The default profile name is based on the profile type and a  
# trailing number, for example:  
# profileType ProfileNumberwhere profileType is a value such as ProcSrv,  
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a  
# unique profile name. The value for this parameter must not contain spaces  
# or characters that are not valid such as the following: *, ?, ", <, >, ,,  
# /, \, and |. The profile name that you choose must not be in use. On  
# Windows platforms: If the fully qualified path contains spaces, enclose  
# the value in quotation marks. The default value is based on the  
# install_root directory, the profiles subdirectory, and the name of the  
# file. For example, the default for profile creation is:  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere  
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties  
# file in the install_root/properties directory. The value for this  
# parameter must be a valid path for the target system and must not be  
# currently in use. You must have permissions to write to the directory.
```

```
#
```

```
# Valid Values:
```

```
# Varies
```

```
# Default Values:
```

```
# None
```

```
#####
```

```
profileName=Custom01
```

```
#####
```

```
# Parameter: cellName
```

```
#
```

```
# Description:
```

```
# Specifies the cell name of the profile. Use a unique cell name for each  
# profile.  
# The default value for this parameter is based on a combination of the  
# short host name, the constant Cell, and a trailing number, for example:  
# if (Dmgr)  
#   shortHostNameCellCellNumber  
# else  
#   shortHostNameNodeNodeNumberCell  
# where CellNumber is a sequential number starting at 01 and NodeNumber is  
# the node number that you used to define the node name.  
# The value for this parameter must not contain spaces or any characters  
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
```

```
#
```

```
# Valid Values:
```

```
# Varies
```

```
# Default Values:
```

```
# Varies
```

```
#
```

```
# Parameter: nodeName
```

```
#
```

```
# Description:
```

```
# Specifies the node name for the node that is created with the new profile.  
# Use a unique value within the cell or on the workstation. Each profile  
# that shares the same set of product binaries must have a unique node name.  
# This parameter is required for profile creation only with the  
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.  
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.  
# The default value for this parameter is based on the short host name,  
# profile type, and a trailing number, for example:
```



```

#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile

```

```

# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you

```

```

# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbType

```

```

#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   ORACLE
# Default Values:
#   ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter:  federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter:  dmgrHost

```

```

#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host

```

```

# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

```

```

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere

```

```

# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01
#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.

```



```

# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux

```

```

# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN
#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the

```

```

# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the
# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dmgrPort
#
# Description:
# Identifies the SOAP port of the deployment manager. Specify this parameter
# and the dmgrHost parameter to federate a custom profile as it is created
# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
# Varies
# Default Values:
# None
#####

```

```

federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter: profileName
#
# Description:

```

```

# Specifies the name of the profile. Use a unique value when creating a
# profile.
# Each profile that shares the same set of product binaries must have a
# unique name. The default profile name is based on the profile type and a
# trailing number, for example:
# profileType ProfileNumber where profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```



```

#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not

```

```

# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.

```

```

# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# ORACLE
# Default Values:
# ORACLE
#
# Parameter: dbDriverType
#
# Description:
# The database driver type. For an Oracle database, specify ORACLE. For
# databases other than Oracle, the value is automatically set based on the
# server operating system. Server installations on all operating systems use
# type 4.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbJDBCClasspath
#
# Description:
# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
# Parameter: federateLaterBPM
#
# Description:
# Indicates if the managed profile is to be federated later using either the
# profileTemplates/managed.esbserver or the
# profileTemplates/BPM/managed.templateName template. Valid values are true
# or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
# -dmgrAdminPassword parameters are not set, the default value for this
# parameter is true. If you use -federateLaterBPM, you must either omit the
# WebSphere Application Server-based parameter -federateLater or set it to
# true.
#
# Valid Values:
# false
# true
# Default Values:
# true
#
# Parameter: dmgrHost
#
# Description:
# Identifies the workstation where the deployment manager is running.
# Specify this parameter and the dmgrPort parameter to federate a custom
# profile as it is created or augmented. This parameter is available with
# the managed.templateName and managed.esbserver profile templates. The
# host name can be the long or short DNS name or the IP address of the

```

```

# deployment manager workstation. Specifying this optional parameter directs
# the manageprofiles command-line utility to attempt to federate the custom
# node into the deployment manager cell as it creates the custom profile.
# This parameter is ignored when creating a deployment manager profile or
# stand-alone server profile. If you federate a custom node when the
# deployment manager is not running, the installation indicator in the logs
# is INSTCONFFAILED to indicate a complete failure. The resulting custom
# profile is unusable. You must move the custom profile directory out of the
# profile repository (the profile's installation root directory) before
# creating another custom profile with the same profile name. If you have
# changed the default JMX connector type, you cannot federate with the
# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.

```

```

# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#

```

```

# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies

```

```

# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr.templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellManagerNodeNumber
#   else
#     shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.

```



```

#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#

```

```

# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter:  dbDriverType
#
# Description:
#   The database driver type. For an Oracle database, specify ORACLE. For
#   databases other than Oracle, the value is automatically set based on the
#   server operating system. Server installations on all operating systems use
#   type 4.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbDriverType=ORACLE_THIN
#####
# Parameter:  federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#

```

```

# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter:  dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####

```

```

# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```

Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-*10-100-75-75-c-60-iso8859-1

```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root*/properties/profileRegistry.xml file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates and under *install_root*/profileTemplates for other products. The following templates are available for custom profiles:

- `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
- `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.

- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
 - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
 - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_Standalone_DB2.response
- PC_Std_Standalone_DB2zOS.response
- PC_Std_Standalone_Oracle.response
- PC_Std_Standalone_SQLServer.response
- PC_Adv_Standalone_DB2.response
- PC_Adv_Standalone_DB2zOS.response
- PC_Adv_Standalone_Oracle.response
- PC_Adv_Standalone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response

- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:
`manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:


```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):


```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

Creating or augmenting network deployment profiles with an SQL Server database server:

You can configure a network deployment environment for IBM Business Process Manager using a Microsoft SQL Server database server.

Preparing the SQL Server database server:

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Restriction: The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 172.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED ON**.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.

- c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
 - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating or augmenting deployment manager profiles:

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles with an SQL Server database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 173. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
	Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating Process Server deployment manager profiles with an SQL Server database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Server using the Profile Management Tool on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default

language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.

- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. **Advanced:** Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

14. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
 - d. Click **Next**.

15. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 174. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
	Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Creating deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available:
 - `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.
 - `dmgr.procctr.adv`: for a Process Center deployment manager profile for IBM BPM Advanced.
 - `dmgr.procsvr`: for a Process Server deployment manager profile for IBM BPM Standard.

- dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response

- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_Standalone_DB2.response
- PS_Std_Standalone_DB2z0S.response
- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2z0S.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadm -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a deployment manager profile using the **manageprofiles** command that includes the following components: CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space.

```
install_root/bin/manageprofiles -create -templatePath install_root\profileTemplates\BPM\
\dmgr.procsvr.adv -dbHostName hostName -dbServerPort 1433 -dbDelayConfig true
-configureBSPACE true -dbType MSSQLSERVER_Microsoft -dbUserId userID -dbJDBCClasspath
path to JDBC 2.0 driver -dbName CMNDB
-dbPassword password
```

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_DMGr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
```



```

# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.

```

```

# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies

```

```

# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you

```

```

# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign

```

```

#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbidbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
# Default Values:
#   false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####

```

```

# Parameter: dbDelayConfig
#
# Description:
# Indicates whether to postpone table creation until after the profile is
# created. Valid values are true or false. The parameter is set to false by
# default. Setting this parameter to true delays the execution of the
# scripts that create the database tables. No. Specify false if you want the
# database to be set up during profile creation.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
# The name of the database. By default, the value is set to orcl for Oracle
# databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# CMNDB
#
# Parameter: dbUserId
#
# Description:
# The user ID for all database types. Specifies the user ID that has
# privileges to create and drop the databases. The WebSphere data source
# uses this ID to authenticate the database connection.
# Important: The -dbUserId parameter value must be a valid database
# authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPassword
#
# Description:
# The password required for database authentication. Required when the
# -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: dbCommonUserId
#

```

```

# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test

```



```

# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .

```

```

#####
#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procctr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#

```

```

# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#

```

```

# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port

```

```

# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   None
#
# Parameter:   winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter:   winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:   winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:   winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:   winserviceStartupType
#
# Description:

```

```

# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbWinAuth
#
# Description:
# An optional parameter that is valid in all default profile templates.
# Specify true to indicate that you are running SQL Server using integrated
# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test

```

```
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false
```

Sample response file for an Advanced Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
```



```

#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter:  templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the

```

```

# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the

```

```

# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the

```

```

# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled

```

```

# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# Default Values:
# false
#####
dbCreateNew=false

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# MSSQLSERVER_MICROSOFT

```

```

# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword

```

```

#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM

```



```

# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify
# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,

```

```

# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/dmgr.procsrv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose

```

```

# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Dmgr01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity

```

```

#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.

```

```

# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile

```

```

# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbWinAuth
#
# Description:
# An optional parameter that is valid in all default profile templates.
# Specify true to indicate that you are running SQL Server using integrated
# windows authentication.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
#dbWinAuth=true
#####
# Parameter: environmentType
#
# Description:
# Specifies the environment type. The environment type refers to how IBM
# Business Process Manager is used; for example, in a production, stage or
# test environment. Load testing might be done on a test server, while a
# stage environment type might be used as a temporary location to host
# changes before putting those changes into production. You might specify

```

```

# Stage as the Environment type if the server you are configuring will be
# accessed and used to review content and new functionality. Valid values
# are as follows:
# Test
# Use Test if the server you are configuring is to be used as a testing
# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Augmenting deployment manager profiles for Process Center with an SQL Server database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```

Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1

```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```


The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.
13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 175. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code>
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.

- Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles for Process Server with an SQL Server database server using the Profile Management Tool:

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
 1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command **install_root/bin/ProfileManagement/pmt.sh**.
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
 8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all

internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

15. Advanced: Configure the Common database using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify the information for the Common database.
 - a. From the **Select a database product** list, select **Microsoft SQL Server**.
 - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 176. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.

Table 176. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
 - Optionally, configure Business Process Choreographer.

Augmenting deployment manager profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install_root/properties/profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root/profileTemplates/BPM* directory for BPM templates and under *install_root/profileTemplates* for other products. The following templates are available:

 - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
 - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
 - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
 - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
 - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the

profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install_root/BPM/samples/manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response

- PS_Adv_DMgr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Dmgr.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Creating or augmenting custom profiles:

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool on 32-bit architectures. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Run the command *install_root/bin/ProfileManagement/pmt.sh*.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
 5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
 6. If you selected **Typical** profile creation, skip to the Federation step.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Creating custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
Templates for each profile are located in the `install_root/profileTemplates/BPM` directory for BPM templates and under `install_root/profileTemplates` for other products. The following templates are available for custom profiles:
 - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.
 - `managed.procctr.adv`: for a Process Center custom profile for IBM BPM Advanced.
 - `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard.
 - `managed.procsvr.adv`: for a Process Server custom profile for IBM BPM Advanced.
 - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the `install_root/BPM/samples/manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2zOS.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`

- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2z0S.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2z0S.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:


```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.sh -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv  
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Response file for creating custom profiles:

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

Sample response file for an Advanced Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT  
# *****  
#  
# Licensed Materials - Property of IBM  
# 5725-C94  
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.  
# This sample program is provided AS IS and may be used, executed, copied  
# and modified without royalty payment by customer (a) for its own  
# instruction and study, (b) in order to develop applications designed to  
# run with an IBM WebSphere product, either for customer's own internal  
# use or for redistribution by customer, as part of such an application,  
# in customer's own products.  
#  
# *****  
# END COPYRIGHT  
  
#####  
# These response file has the applicable parameters for creating a  
# Managed PC Adv profile that uses SQLServer.  
#  
# Depending on your environment, you may need to change the default values.  
#  
# To create a profile with this response file specify:  
#  
# was.install.root/bin/manageprofiles -response PC_Adv_Managed_SQLServer.response  
#  
# If you use the -response parameter, it must be the only parameter on
```

```

# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

```

```

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

```

```

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:

```

```

# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
#####
# Parameter:  dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter:  dbWinAuth
#

```

```

# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM

```



```

# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Center custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
# Specifies the directory path to the template files in the installation
# root directory. Within the profileTemplates directory are various

```

```

# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procctr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the

```

```

#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  enableService
#
# Description:
#    Enables the creation of a Linux service. Valid values include true or
#    false. The default value for this parameter is false. When the
#    manageprofiles command-line utility is run with the -enableService option
#    set to true , the Linux service is created with the profile when the
#    command is run by the root user. When a nonroot user runs the
#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#    Specifies the user ID that is used during the creation of the Linux
#    service so that the Linux service will run under this user ID. The Linux
#    service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#    The type of the owner account of the Windows service created for the
#    profile. Valid values include specifieduser or localsystem. The
#    localsystem value runs the Windows service under the local account of the

```

```

# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword
#
# Description:
# Specify the password for the specified user or the local account that is
# to own the Windows service.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceCheck
#
# Description:
# The value can be either true or false. Specify true to create a Windows
# service for the server process that is created within the profile. Specify
# false to not create the Windows service. The default value for this
# parameter is false.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: winserviceStartupType
#
# Description:
# The following values for Windows service startup can be used:
# manual
# automatic
# disabled
# The default value for this parameter is manual.
#
# Valid Values:
# manual
# automatic
# disabled

```

```

# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the

```

```

# manageprofiles command-line utility. Use the addNode command later after
# the node is created instead. The default value for this parameter is
# localhost. The value for this parameter must be a properly formed host
# name and must not contain spaces or characters that are not valid such as
# the following: *, ?, ", <, >, , , /, \, and |. A connection to the
# deployment manager must also be available in conjunction with the dmgrPort
# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage

```

```

# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####
#####

```



```

# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.

```

```

# The default value for this parameter is based on a combination of the
# short host name, the constant Cell, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellCellNumber
# else
#     shortHostNameNodeNodeNumberCell
# where CellNumber is a sequential number starting at 01 and NodeNumber is
# the node number that you used to define the node name.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter: nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellManagerNodeNumber
#     else
#         shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name
#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in

```

```

# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter: serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux

```

```

#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false

```

```

#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   MSSQLSERVER_MICROSOFT
# Default Values:
#   MSSQLSERVER_MICROSOFT
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort
#   parameter. Required if you are creating a managed profile with the
#   -federateLaterBPM parameter set to false and the deployment manager is not
#   on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created

```

```

# or augmented. The deployment manager must be running and accessible. If
# you have changed the default JMX connector type, you cannot federate with
# the manageprofiles command-line utility. Use the addNode command later
# after the node is created instead. The default value for this parameter is
# 8879. The port that you indicate must be a positive integer and a
# connection to the deployment manager must be available in conjunction with
# the dmgrHost parameter. Required if you are creating a managed profile,
# the -federateLaterBPM parameter is set to false, and the deployment
# manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false

```

```

# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for a Standard Process Server custom profile:

Modify and use this response file to create a custom profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#

```



```

# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/managed.procsvr

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=Custom01

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.

```

```

# Use a unique value within the cell or on the workstation. Each profile
# that shares the same set of product binaries must have a unique node name.
# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#####
cellName=cell_name
nodeName=node_name

#####
# Parameter: enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies

```

```

# Default Values:
#   None
#
# Parameter:   keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:   enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:   serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:   winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser

```

```

# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem

```

```

#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
# Parameter: dbWinAuth
#
# Description:
#   An optional parameter that is valid in all default profile templates.
#   Specify true to indicate that you are running SQL Server using integrated
#   windows authentication.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
#dbWinAuth=true

#####
# Parameter: federateLaterBPM
#
# Description:
#   Indicates if the managed profile is to be federated later using either the
#   profileTemplates/managed.esbserver or the
#   profileTemplates/BPM/managed.templateName template. Valid values are true
#   or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#   -dmgrAdminPassword parameters are not set, the default value for this
#   parameter is true. If you use -federateLaterBPM, you must either omit the
#   WebSphere Application Server-based parameter -federateLater or set it to
#   true.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#
# Parameter: dmgrHost
#
# Description:
#   Identifies the workstation where the deployment manager is running.
#   Specify this parameter and the dmgrPort parameter to federate a custom
#   profile as it is created or augmented. This parameter is available with
#   the managed.templateName and managed.esbserver profile templates. The
#   host name can be the long or short DNS name or the IP address of the
#   deployment manager workstation. Specifying this optional parameter directs
#   the manageprofiles command-line utility to attempt to federate the custom
#   node into the deployment manager cell as it creates the custom profile.
#   This parameter is ignored when creating a deployment manager profile or
#   stand-alone server profile. If you federate a custom node when the
#   deployment manager is not running, the installation indicator in the logs
#   is INSTCONFFAILED to indicate a complete failure. The resulting custom
#   profile is unusable. You must move the custom profile directory out of the
#   profile repository (the profile's installation root directory) before
#   creating another custom profile with the same profile name. If you have
#   changed the default JMX connector type, you cannot federate with the
#   manageprofiles command-line utility. Use the addNode command later after
#   the node is created instead. The default value for this parameter is
#   localhost. The value for this parameter must be a properly formed host
#   name and must not contain spaces or characters that are not valid such as
#   the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#   deployment manager must also be available in conjunction with the dmgrPort

```

```

# parameter. Required if you are creating a managed profile with the
# -federateLaterBPM parameter set to false and the deployment manager is not
# on the local machine.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dmgrPort
#
# Description:
#   Identifies the SOAP port of the deployment manager. Specify this parameter
#   and the dmgrHost parameter to federate a custom profile as it is created
#   or augmented. The deployment manager must be running and accessible. If
#   you have changed the default JMX connector type, you cannot federate with
#   the manageprofiles command-line utility. Use the addNode command later
#   after the node is created instead. The default value for this parameter is
#   8879. The port that you indicate must be a positive integer and a
#   connection to the deployment manager must be available in conjunction with
#   the dmgrHost parameter. Required if you are creating a managed profile,
#   the -federateLaterBPM parameter is set to false, and the deployment
#   manager was configured with a port that was not a default port.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
# Parameter:  environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####

```

```

# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting custom profiles using the Profile Management Tool:

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager on a 32-bit architecture. For 64-bit architectures on Solaris, use the **manageprofiles** command-line utility.

Remember to shut down any servers associated with a profile you plan to augment.

When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```

Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1

```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root/java/bin/java -Duser.language=locale install_root
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root/java/bin/java -Duser.language=de install_root/bin/ProfileManagement/startup.jar
```

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
 2. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.

- Run the command *install_root/bin/ProfileManagement/pmt.sh*.
3. Shut down any servers associated with a profile you plan to augment.
 4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
 9. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root/profiles/profile_name*.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
 10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.

- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root/properties/portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

Related information:

 Updating ports in existing profiles (WebSphere Application Server)

Augmenting custom profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install_root*/properties/profileRegistry.xml file. Do not modify this file; use it only to view the templates.

2. Find the appropriate template to use for the augmentation.

Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates and under *install_root*/profileTemplates for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
- managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
- managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.

3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir*/profileTemplates/BPM directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root*/BPM/samples/manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response

- PC_Std_DMgr_SQLServer.response
- PC_Adv_DMgr_DB2.response
- PC_Adv_DMgr_DB2zOS.response
- PC_Adv_DMgr_Oracle.response
- PC_Adv_DMgr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMgr_DB2.response
- PS_Std_DMgr_DB2zOS.response
- PS_Std_DMgr_Oracle.response
- PS_Std_DMgr_SQLServer.response
- PS_Adv_DMgr_DB2.response
- PS_Adv_DMgr_DB2zOS.response
- PS_Adv_DMgr_Oracle.response
- PS_Adv_DMgr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Use the **chmod** command to assign the appropriate permissions to your new copy of the response file. For example:

```
chmod 644 BPM_Custom.response
```

Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.sh -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv  
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

Federating custom nodes to the deployment manager:

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
 - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:

```
profile_root/bin/startManager.sh
```
 - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
 - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
 - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
 - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile_root* represents the installation location of the custom profile):

```
profile_root/bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username  
userID_for_authentication -password password_for_authentication
```

An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Related information:



Add managed node settings (WebSphere Application Server)

Loading the database with system information in a network deployment environment:

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly

- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. For a cluster, you need to specify the cluster name. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.
- If a single WebSphere cell contains multiple application target clusters, you must run this command on each of the clusters.

Run the bootstrap utility from the command line. The bootstrap utility is found in the deployment manager profile directory. For example:

```
Linux      UNIX      install_root/profiles/Dmgr01/bin
Windows   install_root\profiles\Dmgr01\bin
```

Run the bootstrap utility using one of the following commands:

- **bootstrapProcessServerData.sh -clusterName *cluster_name***
- **bootstrapProcessServerData.sh -nodeName *node_name* -serverName *server_name***

where:

- **-clusterName** is the name of the application target cluster. You must specify this parameter when you want the bootstrap data to run on a cluster.
- **-nodeName** is the name of the node. You must specify this parameter and the **-serverName** parameter when you want the bootstrap data to run on the server that is part of the network deployment environment and not part of the cluster.
- **-serverName** is the name of the server. You must specify either this parameter and the **-nodeName** parameter when you want the bootstrap data to run on the server that is part of the network deployment and not part of the cluster.

The parameters are case-sensitive.

Important: For SQL Server only: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies

this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

You have loaded the database with system information before successfully starting the IBM Business Process Manager server. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.clusterName.timestamp.log` or `bootstrapProcessServerData.nodeName.serverName.timestamp.log` depending on the target you specified. The console displays a subset of the logged information.

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData
```

Bootstrap data onto a cluster that hosts the Process Server or Process Center:

```
bootstrapProcessServerData -clusterName myAppCluster
```

Modifying an existing installation:

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

Installing and configuring IBM Business Process Manager on Windows

Install IBM Business Process Manager on Windows, and configure a stand-alone or network deployment environment.

Configuring a stand-alone environment on Windows

You can use a typical installation to install the software and configure a stand-alone Process Center or Process Server profile. Or you can choose custom installation and then create profiles, giving you more control over how to configure your software.

Typical installation and configuration:

The Typical installation option is the simplest and quickest method for installing and configuring IBM Business Process Manager.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing and configuring IBM Business Process Manager with a new DB2 Express database server:

The typical installation can install DB2 Express on Windows and configure the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (Administrator user) and do not have an existing DB2 database server on the system.

Installing and configuring Process Center with a new DB2 Express database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

The typical installation installs DB2 Express and configures the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (Administrator user) and do not have an existing DB2 database server on the system.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings*user_name*\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users*user_name*\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```


2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Click **Next**.
7. Optionally change the location information.
 - **Hostname:** This field shows the name of your machine.

Important: If a value of `localhost` or `127.0.0.1` is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.

- **Location:** Click **Browse** to change the installation location for Process Center and Process Designer.

Restriction: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

8. Specify the **Username** and **Password** for this server's administrative account.
9. Click **Next**.
10. Select **No** to install DB2 Express.
11. Specify the **Username** and **Password** for the DB2 Express database.

Restriction: User names must not contain National Language Strings (NLS).

12. Click **Next**.
13. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

14. Click **Install Software**.

You have installed a fully functional Process Center and Process Designer.

Note: The Process Designer is supported on the Windows platform only.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 Installation Manager updates

Installing and configuring Process Server with a new DB2 Express database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

The typical installation installs DB2 Express and configures the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (Administrator user) and do not have an existing DB2 database server on the system.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings\user_name\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users\user_name\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=C:\launchpad_updates  
fixpack.WAS_REP=C:\WAS_updates  
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Click **Next**.
7. Specify Process Server information:
 - **Hostname**: This field shows the name of your machine.
 - **Location**: Click **Browse** to change the installation location.

Restriction: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

- Environment **Type**: Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
- Environment **Name**: The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname**: Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port**: Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name**: Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password**: Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

8. Click **Next**.
9. Select **No** to install DB2 Express.

10. Specify the **Username** and **Password** for the DB2 Express database.

Restriction: User names must not contain National Language Strings (NLS).

11. Click **Next**.

12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with a DB2 database server:

You can install IBM Business Process Manager using a DB2 database server. You must make sure the required databases are available before you install.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

Installing and configuring Process Center with a DB2 database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings\user_name\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users\user_name\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Click **Next**.
7. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center and Process Designer.
8. Specify the **Username** and **Password** for this server's administrative account.
9. Click **Next**.
10. Select **Yes** to use an existing database.
11. Specify the required database information.

Table 177. Required database configuration fields for DB2

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 50000 or enter the correct server port number.
Username	Enter the user name to authenticate with the database. Restriction: User names must not contain National Language Strings (NLS).
Password	Enter a password to authenticate with the database.
Common database name	Enter the name for the Common database.
Process Center database name	Enter the name for the Process Center database.
Performance Data Warehouse database name	Enter the name for the Performance Data Warehouse database.

12. Click **Next**.

13. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

14. Click **Install Software**.

You have installed a fully functional Process Center and Process Designer.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with a DB2 database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty. The databases must be created with at least a 32K page size.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed

from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings*user_name*\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users*user_name*\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

(from a command line) *extract_directory*\launchpad.exe

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Click **Next**.
7. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

8. Click **Next**.
9. Select **Yes** to use an existing database.
10. Specify the required database information.

Table 178. Required database configuration fields for DB2

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 50000 or enter the correct server port number.
Username	Enter the user name to authenticate with the database. Restriction: User names must not contain National Language Strings (NLS).
Password	Enter a password to authenticate with the database.
Common database name	Enter the name for the Common database.
Process Server database name	Enter the name for the Process Server database.
Performance Data Warehouse database name	Enter the name for the Performance Data Warehouse database.

11. Click **Next**.
12. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is

available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

13. Click **Install Software**.

You have installed a fully functional Process Server.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 [Installation Manager updates](#)

Installing and configuring IBM Business Process Manager with an Oracle database server:

You can install IBM Business Process Manager using an Oracle database server. You must make sure the required databases are available before you install.

Creating Oracle databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

Installing and configuring Process Center with an Oracle database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings*user_name*\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users*user_name*\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates  
fixpack.2=http://test/rep  
launchpad.1=C:\launchpad_updates  
fixpack.WAS_REP=C:\WAS_updates  
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.
(from a command line) *extract_directory*\launchpad.exe
3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Click **Next**.
7. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center and Process Designer.
8. Specify the **Username** and **Password** for this server's administrative account.
9. Click **Next**.
10. Select **Yes** to use an existing database.
11. Specify the required database information.

Table 179. Required database configuration fields for Oracle

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1521 or enter the correct server port number.
Database name	Enter the correct Process Center database System Identifier.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Center database	For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

12. Click **Next**.

13. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

14. Click **Install Software**.

You have installed a fully functional Process Center and Process Designer.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with an Oracle database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings*user_name*\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users*user_name*\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

(from a command line) *extract_directory\launchpad.exe*

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - Environment **Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - Environment **Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy

server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Click **Next**.
8. Select **Yes** to use an existing database.
9. Specify the required database information.

Table 180. Required database configuration fields for Oracle

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1521 or enter the correct server port number.
Database name	Enter the correct Process Center database System Identifier.
Common database	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Server database	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database.
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.

Table 180. Required database configuration fields for Oracle (continued)

Field	Action needed
Service Component Architecture application bus messaging engine database	<p>For the SCA application bus messaging engine, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

10. Click **Next**.

11. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

12. Click **Install Software**.

You have installed a fully functional Process Server.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 Installation Manager updates

Installing and configuring IBM Business Process Manager with an SQL Server database server:

You can install IBM Business Process Manager using a Microsoft SQL Server database server. You must make sure the required databases are available before you install.

Creating SQL Server databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must make sure the databases are available before you install IBM Business Process Manager using a typical installation.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
 - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

Important: Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
    at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmntExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.

- d. Click the **Security** tab on the **Local DTC Properties** window.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the Properties window, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
 - b. Extract the archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Run the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Installing and configuring Process Center with an SQL Server database server:

Process Center includes a repository for all processes, services, and other assets created in the IBM Business Process Manager authoring environments. You can use the integrated Process Server within Process Center to run processes as you build them. When you are ready, you can install and run those same processes on the Process Server in your runtime environments.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings*user_name*\

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users\user_name\

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

(from a command line) *extract_directory\launchpad.exe*

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Center**.
6. Click **Next**.
7. Optionally change the location information:
 - **Hostname:** This field shows the name of your machine.

Important: If a value of localhost or 127.0.0.1 is used for the hostname, Process Designer installations on a remote system will not be able to connect to the Process Center.
 - **Location:** Click **Browse** to change the installation location for Process Center and Process Designer.
8. Specify the **Username** and **Password** for this server's administrative account.
9. Click **Next**.
10. Select **Yes** to use an existing database.
11. Specify the required database information.

Table 181. Required database configuration fields for SQL Server

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1433 or enter the correct server port number.
Select the Apply Windows authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the remaining fields are made inactive.	

Table 181. Required database configuration fields for SQL Server (continued)

Field	Action needed
Common database name	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Center database name	For the Process Center database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Center database user name. • Password: Enter a password to authenticate with the Process Center database.
Performance Data Warehouse database name	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

12. Click **Next**.

13. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet

access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

14. Click **Install Software**.

You have installed a fully functional Process Center and Process Designer.

Note: The Process Designer is supported on the Windows platform only.

Related information:

 Installation Manager updates

Installing and configuring Process Server with an SQL Server database server:

Process Server provides a single BPM runtime environment that can support a range of business processes for test, staging, or production.

To install IBM Business Process Manager Advanced using the typical installation, you must confirm that the Process Center or Process Server database, Performance Data Warehouse database, and Common database already exist and that they are empty.

Be prepared to specify the following information during installation:

- The user name and password for database authentication
- The database server host name and port
- The name of the Process Center or Process Server database.
- The name of the Performance Data Warehouse database.
- The name of the Common database.

The typical installation installs the software and configures a stand-alone profile. You do not need to create a profile later using the Profile Management Tool or **manageprofiles** command.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the latest refresh pack and interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

```
On Windows XP, the HOMEPATH environment variable points to C:\Documents and Settings\user_name\
```

```
On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to C:\Users\user_name\
```

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses three prefixes: ifix, fixpack, and launchpad. Each prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifixes, fixpacks, and launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

2. Start the launchpad. You can run only one launchpad at a time.

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

3. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically. If you do not have access to the Internet and want the updates to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates and which updates to install, using the launchpad prefix as described in step 1.
4. After starting the launchpad, click **Typical installation** on the Welcome page.
5. Select **Install Process Server**.
6. Specify Process Server information:
 - **Hostname:** This field shows the name of your machine.
 - **Location:** Click **Browse** to change the installation location.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.
 - **Environment Name:** The environment name is used to connect from a Process Center to this Process Server.

Restriction: Do not mix production and non-production servers in the same cell.

- Specify the **Username** and **Password** for this server's administrative account.

Select **Use this server offline** if you did not select **Production** as the environment **Type** or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline Process Server differs from the method for deploying process applications to an online Process Server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Hostname:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.

- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

7. Click **Next**.
8. Select **Yes** to use an existing database.
9. Specify the required database information.

Table 182. Required database configuration fields for SQL Server

Field	Action needed
Hostname	Accept the default value of localhost or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1433 or enter the correct server port number.
Select the Apply Windows authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the remaining fields are made inactive.	
Common database name	For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common database user name. • Password: Enter a password to authenticate with the Common database.
Process Server database name	For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Process Server database user name. • Password: Enter a password to authenticate with the Process Server database.
Performance Data Warehouse database name	For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Performance Data Warehouse database user name. • Password: Enter a password to authenticate with the Performance Data Warehouse database.
Common Event Infrastructure bus messaging engine database	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the Common Event Infrastructure bus messaging engine user name. • Password: Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
Service Component Architecture system bus messaging engine database	For the SCA system bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA system bus messaging engine user name. • Password: Enter a password to authenticate with the SCA system bus messaging engine.
Service Component Architecture application bus messaging engine database	For the SCA application bus messaging engine, enter values for the following parameters: <ul style="list-style-type: none"> • User name: Enter the SCA application bus messaging engine user name. • Password: Enter a password to authenticate with the SCA application bus messaging engine.

10. Click **Next**.

11. On the Installation summary page, verify the installation options and read the license agreements. If you agree to the terms of the license agreements, click **I have read and accepted the license agreement and notices**. If you select the **Use your support account to include updates with the installation** option, you must provide your IBM User Name and Password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: You must install the mandatory interim fix for APAR JR44669 as part of the typical installation and configuration. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. If you do not have Internet access during installation, download the mandatory interim fix for APAR JR44669 so that it is available during installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the additional required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the remaining required fixes.

12. Click **Install Software**.

You have installed a fully functional Process Server.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 [Installation Manager updates](#)

Custom installation and configuration:

Use the Custom installation option to install IBM Business Process Manager if you need any installation or configuration options that are not provided by the Typical installation option, if you want to install silently, or if you want to install on an existing installation of WebSphere Application Server.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced:

With custom installation, you can select interactive or silent installation. You must select interactive installation if you want to install on an existing WebSphere Application Server.

Related reference:

 [IBM Business Process Manager Advanced system requirements](#)

Installing IBM Business Process Manager Advanced interactively:

You can install the IBM Business Process Manager product package interactively. You must choose interactive installation if you want to install on an existing WebSphere Application Server.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related reference:



IBM Business Process Manager Advanced system requirements

Installing interactively with a new installation of WebSphere Application Server:

Using custom installation, you can install IBM Business Process Manager with a new installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `DVD_root\launchpad.exe`

- If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

`C:\HOMEPATH\bpm_updates.properties`

On Windows XP, the HOMEPATH environment variable points to `C:\Documents and Settings\user_name\`

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to `C:\Users\user_name\`

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Install as administrative user** to install as an administrative user. If you are in the Administrators group, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.
5. Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
6. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package.
7. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

8. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**.

Tip: Keep the installation path as short as possible. Otherwise, you might run into problems later when the paths of other components, when added to this path, exceed the 255-character path limit. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
12. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:

 [Installation Manager updates](#)

Installing interactively with a new DB2 Express database and a new installation of WebSphere Application Server:

The custom installation can install DB2 Express on Windows and configure the required databases for IBM Business Process Manager. Only select this path if you have administrative privileges (Administrator user) and do not have an existing DB2 database server on the system.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

(from a command line) `DVD_root\1launchpad.exe`
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:

To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

(from a command line) `extract_directory\launchpad.exe`

- Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to `C:\Documents and Settings\user_name\`

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to `C:\Users\user_name\`

Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

- After starting the launchpad, click **Custom installation** on the Welcome page.
- Click **Install as administrative user** to install as an administrative user. If you are in the Administrators group, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.
- Click **Install**. When you install IBM Business Process Manager, the required WebSphere Application Server Network Deployment is automatically installed.
- On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Select IBM DB2 Express to install and use an embedded DB2 Express database.
- Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

- On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.

9. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**.

Tip: Keep the installation path as short as possible. Otherwise, you might run into problems later when the paths of other components, when added to this path, exceed the 255-character path limit. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Attention: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

10. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

11. Optional: Specify the DB2 administrative **User name** and **Password** on the Common Configurations page. The default administrative user name is bpmadmin. Choose a password that complies with the password policy of your system (for example, Windows 2008).

Click **Re-Validate** to confirm that your choices are correct.

Restriction: User names must not contain National Language Strings (NLS).

12. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
13. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:

 Installation Manager updates

Installing interactively on an existing installation of WebSphere Application Server:

Using custom installation, you can install IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.
- If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

(from a command line) `DVD_root\launchpad.exe`

(from a command line) `extract_directory\launchpad.exe`

2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

`C:\HOMEPATH\bpm_updates.properties`

On Windows XP, the HOMEPATH environment variable points to `C:\Documents and Settings\user_name\`

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to `C:\Users\user_name\`

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Installation on an existing WebSphere Application Server**.

5. Click **Install as administrative user** to install as an administrative user. If you are in the Administrators group, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.
6. Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
7. Click **Install**.
8. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server.
9. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

10. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
11. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.
12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back**

and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.

14. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

Related information:



Installation Manager updates

Installing interactively with a new DB2 Express database on an existing installation of WebSphere Application Server:

Using custom installation, you can install DB2 Express and IBM Business Process Manager on an existing installation of WebSphere Application Server.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage. You can run only one launchpad at a time.
 - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Advanced into the disk drive. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter the following command to start the launchpad manually:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.
(from a command line) `DVD_root\launchpad.exe`
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter the following command to start the launchpad:
To install or run IBM Business Process Manager Advanced on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.
(from a command line) `extract_directory\launchpad.exe`
2. Optional: If you see a message that prompts you to update the launchpad, click **Update** to receive the latest updates. The updates are installed and your launchpad is restarted automatically.

If you do not have access to the Internet and want updates to the launchpad to be installed from a local directory, you can use a properties file to tell the Installation Manager where to find the updates. Create the following file:

```
C:\HOMEPATH\bpm_updates.properties
```

On Windows XP, the HOMEPATH environment variable points to `C:\Documents and Settings\user_name\`

On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to `C:\Users\user_name\`

Note: Ensure that you have read/write access to the folders specified in the bpm_updates.properties file.

The file uses a launchpad prefix, followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for launchpad upgrades. The locations can be either local directories or URLs. For example:

```
ifix.1=C:\bpmUpdates
fixpack.2=http://test/rep
launchpad.1=C:\launchpad_updates
fixpack.WAS_REP=C:\WAS_updates
fixpack.BPM_REP=C:\BPM_updates
```

3. After starting the launchpad, click **Custom installation** on the Welcome page.
4. Click **Installation on an existing WebSphere Application Server**.
5. Click **Install as administrative user** to install as an administrative user. If you are in the Administrators group, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.
6. Click **Update** to update WebSphere Application Server. When Installation Manager opens, click **Update** to install available updates. On the Update Packages page, select **Show all** to display available updates.
7. Click **Install**.
8. On the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Advanced** package. Clear the check box beside WebSphere Application Server. Select IBM DB2 Express to install and use an embedded DB2 Express database. The user must have administrative privileges to install DB2 Express.
9. Click **Next** to continue. You are prompted to provide your IBM ID and password to connect to the IBM service repositories.

The connection to the service repositories is required to download and install any fix packs and required interim fixes from the Internet, including fixes for WebSphere Application Server and IBM Business Process Manager. An IBM ID and password can be obtained by registering at <http://www.ibm.com>.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly for databases configured as part of this installation. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

You can click **Cancel** to continue installing without downloading the required fixes from the Internet. After successfully installing the product, you can use Installation Manager to install the required fixes. See Installing the required fixes for the IBM Business Process Manager V8.0 products and IBM Business Monitor V8.0 for more information about downloading and installing the required fixes.

10. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.
11. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

If you see a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. See the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Attention: Because you are installing DB2 Express, your installation location cannot contain National Language Strings (NLS).

12. On the Features page, expand the plus symbol to select the package features that you want to install. Installation Manager automatically enforces any dependencies with other features and shows the updated download size and disk space requirements for the installation.
 - a. Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - b. Optional: Click a feature to view its brief description under **Details**.
 - c. Optional: If you are installing Process Server, you can expand the Process Server feature and change the Process Server use. Select "Production" (the default) to use the server in production, or "Non-production" to use the server only for test, staging, or development. Your selection is recorded in the product tag for inventory purposes.

Restriction: Do not mix production and non-production servers in the same cell.

When you are finished selecting features, click **Next**.

13. Optional: Specify the DB2 administrative **User name** and **Password** on the Common Configurations page. The default administrative user name is bpmadmin. Choose a password that complies with the password policy of your system (for example, Windows 2008).
Click **Re-Validate** to confirm that your choices are correct.

Restriction: User names must not contain National Language Strings (NLS).

14. On the Summary page, review your choices before installing the IBM Business Process Manager Advanced package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
15. When the installation process is complete, a message confirms the success of the process.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to run the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

If you used the Windows domain user ID to create a new DB2 Express database and tables, and profile creation failed, use one of the following solutions:

- Log into the Windows system as a local system user and run the Profile Management Tool to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" and then create a profile.

Related information:



Installation Manager updates

Installing IBM Business Process Manager silently:

You can install the IBM Business Process Manager product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Before you install IBM Business Process Manager, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

Related reference:

IBM Business Process Manager Advanced system requirements

Installing silently using the command line:

You can install IBM Business Process Manager silently using the command line. With this method, you only need to run one command to perform the installation.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

```
extract_directory\IM\tools\imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.
3. Run the following command:

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

```
extract_directory\IM\tools\imcl install list_of_product_IDs -acceptLicense  
-installationDirectory location -repositories repository  
-properties key=value,key=value -showVerboseProgress -log logName.log
```

where:

- *list_of_product_IDs* is a list of the IDs for the products and features that you want to install. You must include the required features. The syntax is *productID,feature,feature*, with multiple products separated by spaces.

Table 183. Product IDs

Product	Product ID	Feature and Description
IBM BPM Advanced	com.ibm.bpm.ADV.V80	<ul style="list-style-type: none"> • bpmAdv.nonprod: Test, staging, or development use. You must specify either this feature or bpmAdv.prod. • bpmAdv.prod: Production use. You must specify either this feature or bpmAdv.nonprod. • wps.client.feature: Required. Run client applications that interact with a IBM BPM Advanced: Process Server in the same cell. • wps.server.feature: Required. Full installation of IBM BPM Advanced: Process Server, which includes the client code and the server code.
WebSphere Application Server Network Deployment	com.ibm.websphere.ND.v80	<ul style="list-style-type: none"> • core.feature: Required. WebSphere Application Server core content. • ejbdeploy: Required. Pre-Enterprise JavaBeans (EJB) 3.0 modules. • thinclient: Required. Stand-alone thin clients and resource adapters. • embeddablecontainer: Embeddable EJB container. • samples: Sample applications feature. • com.ibm.sdk.6_32bit: 32-bit software development kit (SDK). You must specify either this feature or com.ibm.sdk.6_64bit. • com.ibm.sdk.6_64bit: 64-bit SDK. This feature can only be selected on a 64-bit system. You must specify either this feature or com.ibm.sdk.6_32bit.
Installation Manager	com.ibm.cic.agent	<ul style="list-style-type: none"> • agent_core: Installation Manager core content. • agent_jre: Installation Manager Java Runtime Environment (JRE).
DB2 Express for Windows 32-bit	com.ibm.ws.DB2EXP97. winia32	<ul style="list-style-type: none"> • n/a: DB2 must match the operating system and bitness. • n/a: DB2 must match the operating system and bitness.

- *location* is the path to the directory where you want to install the products.

Tip: Keep the installation path as short as possible. Otherwise, you might run into problems later when the paths of other components, when added to this path, exceed the 255-character path limit.

- *repository* is the path to the repository where you have extracted the files, one of the following directories:

```
extract_directory\repository\repos_32bit
extract_directory\repository\repos_64bit
```

For more than one repository, separate the repository locations with commas.

- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas. Create encrypted passwords using the IBM Installation Manager.

Table 184. Keys

Key	Description
user.db2.admin.username	User name with authority to access the DB2 database. The default value is bpmadmin.
user.db2.admin.password	Encrypted password for the user name with authority to access the DB2 database. Choose a password that complies with the password policy of your system (for example, Windows 2008).
user.db2.use.existing	If you are using an existing DB2 database server, add the following line exactly: user.db2.use.existing=true If DB2 must be installed, specify the value <i>false</i> .
user.db2.port	Port for the DB2 database. The default value is 50000.

- *logName* is the name of the log file to record messages and results.

Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for imcl.

Installation Manager installs the list of products and writes a log file to the directory that you specified. The log file is empty if there are no errors or warnings.

The following example installs IBM Business Process Manager Advanced, WebSphere Application Server Network Deployment, and DB2 Express on Windows.

```
imcl install com.ibm.bpm.ADV.V80,bpmAdv.prod,wps.client.feature,wps.server.feature
com.ibm.websphere.ND.v80,core.feature,ejbdeploy,thinclient,embeddablecontainer,samples,com.ibm.sdk.6_32bit
com.ibm.ws.DB2EXP97.winia32 -acceptLicense
-installationDirectory C:\IBM\BPM80 -repositories D:\temp\BPM\repository
-properties user.db2.admin.username=bpmadmin,user.db2.admin.password=Vvrs88V/a9BUdxwodz0nUg==
-showVerboseProgress -log silentinstall.log
```

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

Related reference:

 [Command-line arguments for imcl](#)

 [IBM Business Process Manager Advanced system requirements](#)

Related information:

 [IBM WebSphere Application Server Information Center](#)

Installing silently using a response file:

You can install IBM Business Process Manager Advanced silently by creating a response file and then running a command to use that response file to install the product.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Advanced installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment, including the `ejbdeploy` and `thinclient` features.

By using response files, you can simplify the silent installation and reduce error in the process because you set up your installation options once in a saved, sharable file that can be used on one or more machines. The installation software provides sample response files for each supported operating system and bit version. You can use an unmodified sample response file to perform a silent installation using default settings, or you can modify the response file to set particular values. The comments in the sample response files provide detailed instructions and information about setting the values.

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Advanced.

Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager.

1. Run the following command to generate encrypted passwords using IBM Installation Manager to securely connect to DB2 and the administrative console.

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

```
extract_directory\IM\tools\imutilsc -silent -nosplash encryptString password_to_encrypt
```

2. Create the response file that will install the required base products and IBM Business Process Manager Advanced. Copy a sample response file, suitable for your bit version and user access level, from the following directory:

```
extract_directory\responsefiles\BPM\
```

Alternatively, you can create a response file by recording your actions in Installation Manager. When you record a response file, the selections that you make in Installation Manager are stored in an XML file. When you run Installation Manager in silent mode, Installation Manager uses the data in the XML response file to perform the installation.

3. The default values provided in the sample response files will perform a basic install, but you should review the file and its comments, and modify the parameters as needed for your environment and the access level of your user ID.

In particular, review the following parameters and values:

- For non-Administrator user IDs, check that all location variables point to locations that the user has permissions for.
- Repository location. If you are not running directly from the `extract_directory\responsefiles\BPM\` directory, point to the location of your installation repository. The repository can be local or remote. If you are running from the DVD, copy the response file from the DVD and point it back at the repository on the DVD.
- Installation location (**installLocation**) where IBM Installation Manager is already installed or will be installed.
- Installation location where IBM Business Process Manager Advanced will be installed.

Tip: Keep the installation path as short as possible. Otherwise, you might run into problems later when the paths of other components, when added to this path, exceed the 255-character path limit.

- Eclipse location (**eclipseLocation**).
 - The list of features for the product.
 - Production or non-production use.
 - If you are installing DB2 Express, follow the instructions in the response file for providing the necessary user IDs and passwords. Use the instructions in step 1 for generating the encrypted passwords to include.
4. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.
 5. Run the following command:

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

Administrator:

```
extract_directory\IM\installc.exe -acceptLicense input  
  extract_directory\responsefiles\BPM\response_file_name.xml  
  -log preferred_log_location\silent_install.log
```

Nonadministrator:


```
extract_directory\IM\userinstc.exe -acceptLicense input  
  extract_directory\responsefiles\BPM\response_file_name.xml  
  -log preferred_log_location\silent_install.log
```

Installation Manager installs any required prerequisites and IBM Business Process Manager Advanced, and writes a log file to the directory you specified.

Important: Ensure that you install the mandatory interim fix for APAR JR44669 before you configure and load data into the databases. Otherwise, future interim fixes, fix packs, or refresh packs might not install properly. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

Then, define a stand-alone server or deployment manager profile in the Profile Management Tool or using the **manageprofiles** command-line utility. Only profiles created with the Profile Management Tool or **manageprofiles** can be used in production.

Related tasks:

 Installing silently by using Installation Manager

 Recording a response file with Installation Manager

Related reference:

 IBM Business Process Manager Advanced system requirements

Granting write permission of files and directories to nonadministrative users for profile creation or augmentation:

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be an administrative or nonadministrative user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonadministrative users. The nonadministrative users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonadministrative users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonadministrative users must store their profiles in their private directory structure, not in the *install_root*\profiles directory of the product.

Restrictions:

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonadministrative users. A nonadministrative user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonadministrative users. The nonadministrative user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonadministrative users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the *profilers* group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a administrative or nonadministrative user.
2. Using operating system commands, perform the following steps:
 - a. Create a group named **profilers**, which will contain all users who can create profiles.
 - b. Create a user named **user1**, who can create profiles.
 - c. Add users **product_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
 - Create the *install_root*\logs\manageprofiles directory by following instructions in the Windows documentation. For this example procedure, the directory is:
install_root\logs\manageprofiles
 - Create the *install_root*\properties\fsdb directory by following instructions in the Windows documentation. For this example procedure, the directory is:
install_root\properties\fsdb
5. If no profile exists, create the profileRegistry.xml file as the installer. For this example, the file path is:
install_root\properties\profileRegistry.xml

Add the following information to the profileRegistry.xml file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **C:\Program Files\IBM\WebSphere\AppServer**. Follow instructions in the Windows documentation to give the **profilers** group read and write permission to the following directories and their files:

```
@WASHOME\logs\manageprofiles
@WASHOME\properties
@WASHOME\properties\fsdb
@WASHOME\properties\profileRegistry.xml
```

You might have to change the permissions on additional files if the nonadministrative user encounters permission errors. For example, if the product installer authorizes a nonadministrative user to delete a profile, then the product installer might have to delete the following file:

```
install_root\properties\profileRegistry.xml_LOCK
```

Give write access to the nonadministrative user for the file to authorize the user to delete the file. If the nonadministrative user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonadministrative user to create profiles.

The nonadministrative user that belongs to the **profilers** group can create profiles in a directory that the nonadministrative user owns and to which the nonadministrative user has write permission. However, the nonadministrative user cannot create profiles in the installation root directory of the product.

The administrative user and the nonadministrative user can use the same tasks to manage profiles.

Configuring profiles and databases:

After you install the product, if you did not use typical installation, you must create or augment one or more profiles to define the runtime environment. Before starting a profile, you must have configured the databases that are to be used with the profile.

Configuring a stand-alone environment with a DB2 database server on Windows:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 database server.

Creating DB2 databases:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

Creating database design files for DB2:

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Creating database design files for stand-alone profiles with DB2:

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
    When a db_scripts_output_directory is given, the database scripts
    in the specified directory will be validated. Currently only
    scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
  [db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
    The generated scripts will be put in the corresponding output
    directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5)wesb.nd.topology
(6)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB]      WBI_CommonDB : [master] [status = not complete]
(2)[BPC]              WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE]          WBI_BSPACE : [status = not complete]
(6)[SibME]           WBI_BPC_ME : [status = not complete]
(7)[SibME]           WBI_CEI_ME : [status = not complete]
(8)[SibME]           WBI_SCA_APP_ME : [status = not complete]
(9)[SibME]           WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe]          BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe]          BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :

- At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

- To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

[info] Please pick one of the following [database provider(s)] :

(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=\${WAS_INSTALL_ROOT}/jdbcd drivers/DB2] :
Universal JDBC driver path[default=\${WAS_INSTALL_ROOT}/jdbcd drivers/DB2] :

Note: The password is encrypted in the generated output files.

- At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI_CommonDB is complete with 0 remaining item(s):

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :


```

(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]

```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the master component, all database components are listed as complete.

[info] Please pick one of the following [database component(s)] :

```

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]

```

Please enter the number for the database component :5

[status] bpm.standard.standalone is complete with 0 remaining item(s):

- When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

- Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```

[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB

```

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Creating database design files for specific components with DB2 in a stand-alone environment:

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

-? , -help
display help info.

-e db_design_file_name
edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
when a db_design_file is given, validation will be done on the specified database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :

(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commond
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

Tip: The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Database User name[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

Regular pagesize[default=32k] :

Regular TableSpace[default=TWSSPACE] :

Temporary pagesize[default=32k] :

Temporary TableSpace[default=TWSTEMPSPACE] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=50000] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating or augmenting stand-alone profiles with a DB2 database server:

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

Preparing the DB2 database server:

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

If you plan to configure the databases on a remote server, you must have created the databases before beginning to create or augment the profile. You can create databases on the local server or use existing databases on a remote server.

Creating Process Center stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.

- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

Restrictions: Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
 - Run the command `install_root\bin\ProfileManagement\pmt.bat`
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
 - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
 - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
 - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.

- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root\properties\portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 185. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
20. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEMDB. If these conditions are too

restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating Process Server stand-alone profiles with a DB2 database server using the Profile Management Tool:

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the *install_root\bin\ProfileManagement* directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the

install_root\bin\ProfileManagement\pmt.bat file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
 - Run the command *install_root*\bin\ProfileManagement\pmt.bat
 2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
 4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
 5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
 6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
 7. Advanced: On the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
 - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install_root*\profiles\profile_name.
 - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
 - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
- Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.

- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case.

To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
 - a. Select **Create a web server definition**.
 - b. Specify the web server characteristics on the page, and click **Next**
 - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
 - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 186. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in \${WAS_INSTALL_ROOT}/db2/java.
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.

21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
 - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
 - Optionally, configure Business Process Choreographer.
 - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Creating stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Restriction: To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
 - Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install_root*\profileTemplates\BPM directory for BPM templates and under *install_root*\profileTemplates for other products. The following templates are available:

- default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
- default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.

- default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
 3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2zOS.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2zOS.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2zOS.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2zOS.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2zOS.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2zOS.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2zOS.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2zOS.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response

- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2zOS.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2zOS.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2zOS.response
- PS_Std_StandAlone_Oracle.response
- PS_Std_StandAlone_SQLServer.response
- PS_Adv_StandAlone_DB2.response
- PS_Adv_StandAlone_DB2zOS.response
- PS_Adv_StandAlone_Oracle.response
- PS_Adv_StandAlone_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

Note: The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Response file for creating stand-alone profiles:

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
```

```

#####
templatePath=BPM/default.procctr.adv

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```



```

#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#
# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.

```

```

#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:

```

```

# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem
#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:

```

```

# manual
# automatic
# disabled
# Default Values:
# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER

```

```

#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName

```

```

#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```



```

#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter:  perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
# Parameter:  dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrMePassword
#

```

```

# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
#   Performance Data Warehouse Messaging Engine user name. Required for
#   setting up the Performance Data Warehouse Messaging Engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPerfDWMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
#   The schema name for the Common database. If no value is specified, the
#   following default database schema names are used:
#   DB2 database
#   The default schema name is the value specified by the dbUserId parameter.
#   The -cdbSchemaName parameter is valid for all database types except for
#   Oracle and Microsoft SQL Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCommonUserId
#
# Description:
#   The user id to authenticate with the Common database.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None

```

```

#
# Parameter: dbCommonPassword
#
# Description:
#   The password for the user id of the dbCommonUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name

#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing

```

```

# environment.
# Stage
# Use Stage if the server is to serve as a staging platform to be used as a
# preproduction server.
# Production
# Use Production if the server is to serve in a production capacity.
# The default value is Test.
#
# Valid Values:
# Test
# Production
# Stage
# Default Values:
# Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
# Specifies whether the server is intended for development purposes only.
# This parameter is useful when creating profiles to test applications on a
# non-production server prior to deploying the applications on their
# production application servers. If -isDeveloperServer is set when
# creating an IBM Business Process Manager profile, then a preconfigured VMM
# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

#####
# Parameter: webFormConfig
#
# Description:
# Indicates if Business Space is configured to use IBM Forms Server to work
# with Human Task Management widgets. The default value for this parameter
# is false. Indicate true to configure Business Space to use IBM Forms
# Server. Both the webFormConfig and webFormInstallRoot parameters are
# required to configure IBM Forms Server. This parameter is valid for
# stand-alone server profiles only.
# Note: IBM Forms Server configuration using these parameters is only valid
# for local IBM Forms Server installations.
#
# Valid Values:
# false
# true
# Default Values:
# false
#
# Parameter: webFormHTTP
#
# Description:
# Specifies the location of the IBM Forms Server Translator. The default URL
# for the location is http://localhost:8085/translator. No. This parameter
# is valid only if the -webFormConfig parameter is set to true. It is valid
# for stand-alone server profiles only.
#
# Valid Values:
# Varies

```

```

# Default Values:
#   None
#
# Parameter: webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be
#   C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#   if the -webFormConfig parameter is set to true. It is valid for
#   stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
# Parameter: configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false

#####
# Parameter: configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#####
configureBPC=true

```

Sample response file for a Standard Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procctr

#####
```



```

# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcCtr01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (Dmgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:

```

```

#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter:  signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:

```

```

# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter: defaultPorts
#
# Description:
# Assigns the default or base port values to the profile.d o not use this
# parameter when using the -startingPort or -portsFile parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: startingPort
#
# Description:
# Specifies the starting port number for generating and assigning all ports
# for the profile.
# Port values are assigned sequentially from the -startingPort value,
# omitting those ports that are already in use. The system recognizes and
# resolves ports that are currently in use and determines the port
# assignments to avoid port conflicts.
# Do not use this parameter with the -defaultPorts or -portsFile parameters.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#
# Parameter: portsFile
#
# Description:
# An optional parameter that specifies the path to a file that defines port
# settings for the new profile.
# Do not use this parameter when using the -startingPort or -defaultPorts
# parameter.
# During profile creation, the manageprofiles command-line utility uses an
# automatically generated set of recommended ports if you do not specify the
# -startingPort parameter, the -defaultPorts parameter or the -portsFile
# parameter. The recommended port values can be different than the default
# port values based on the availability of the default ports.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter: enableService
#
# Description:
# Enables the creation of a Linux service. Valid values include true or
# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is

```

```

#    localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter: winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter: winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter: winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:

```

```

# manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
# Specifies the file that holds the database configuration design for all of
# the IBM Business Process Manager components .
# This parameter replaces the -wbldbDesign that was used in previous
# versions of WebSphere Process Server.
#
# Note: If a bpmdbDesign file parameter is specified, none of the database
# related properties should be specified.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
# Indicates a new database is created or if an existing database is to be
# reused. Valid values are true or false. This value cannot be true for an
# Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
# a DB2 database.
#
# Valid Values:
# false
# true
# Default Values:
# None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
# The database type. Set one of the following values for the type of
# database product you are using with IBM Business Process Manager.
#
# Required only for one of the following:
# A stand-alone profile when -bpmdbDesign was not specified.
# An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# DB2_DATASERVER
# Default Values:
# DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:

```

```

# The directory path location of JDBC driver files.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.

```



```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#

```

```

# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#    Varies
# Default Values:
#    None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter:  dbPerfDWMeUserId
#
# Description:
#    Performance Data Warehouse Messaging Engine user name. Required for
#    setting up the Performance Data Warehouse Messaging Engine.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPerfDWMePassword
#
# Description:
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter:  cdbSchemaName
#
# Description:
#    The schema name for the Common database. If no value is specified, the
#    following default database schema names are used:
#    DB2 database
#    The default schema name is the value specified by the dbUserId parameter.
#    The -cdbSchemaName parameter is valid for all database types except for
#    Oracle and Microsoft SQL Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonUserId
#
# Description:
#    The user id to authenticate with the Common database.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:

```

```

# The password for the user id of the dbCommonUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test
#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM

```

```

# file repository is installed. This file repository contains a sample
# organization that can be used to test Business Process Choreographer
# people resolution, ready for you to use as is.
#
# Valid Values:
# false
# true
# Default Values:
# false
#####
#isDeveloperServer=false

```

Sample response file for an Advanced Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94
# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
# Creates the profile.
# If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
# Specifies the directory path to the template files in the installation
# root directory. Within the profileTemplates directory are various

```

```

# directories that correspond to different profile types and that vary with
# the type of product installed. The profile directories are the paths that
# you indicate while using the -templatePath option.
# Use absolute paths. This parameter must exist as a directory and point to
# a valid template directory.
#
# When using the -templatePath parameter, specify the fully qualified file
# path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsrv.adv

#####
# Parameter:  profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:
#   profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#   Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#   unique profile name. The value for this parameter must not contain spaces
#   or characters that are not valid such as the following: *, ?, ", <, >, ,,
#   /, \, and |. The profile name that you choose must not be in use. On
#   Windows platforms: If the fully qualified path contains spaces, enclose
#   the value in quotation marks. The default value is based on the
#   install_root directory, the profiles subdirectory, and the name of the
#   file. For example, the default for profile creation is:
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#   WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#   file in the install_root/properties directory. The value for this
#   parameter must be a valid path for the target system and must not be
#   currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter:  applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter:  cellName
#

```

```

# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellCellNumber
#   else
#       shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.
#   This parameter is required for profile creation only with the
#   dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#   procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#   The default value for this parameter is based on the short host name,
#   profile type, and a trailing number, for example:
#   if (DMgr)
#       shortHostNameCellManagerNodeNumber
#   else
#       shortHostNameNodeNodeNumber
#   where NodeNumber is a sequential number starting at 01.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter:  serverName
#
# Description:
#   Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
# Parameter:  processCenterURL
#
# Description:
#   Specifies the URL of the Process Center that the Process Server connects
#   to. If this parameter is set, the protocol must be either http or https.
#   Note: No corresponding useServerOffline parameter exists to use the server
#   offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  processCenterUserId

```



```

#
# Description:
#   Specifies the user ID to be used to connect to the Process Center. This
#   parameter is valid for stand-alone Process Server profiles. This parameter
#   is required if the Process Server is online (connected to a Process
#   Center). It is required for a stand-alone process server (standard or
#   advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: processCenterPassword
#
# Description:
#   Specifies the user password to be used to connect to the Process Center.
#   This parameter is valid for stand-alone Process Server profiles. This
#   parameter is required if the Process Server is online (connected to a
#   Process Center). It is required for a stand-alone process server (standard
#   or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
# Parameter: enableAdminSecurity
#
# Description:
#   For IBM Business Process Manager omit this parameter. Administrative
#   security is always enabled for IBM Business Process Manager profiles.
#   You must also specify the parameters -adminUserName and -adminPassword
#   along with the values for these parameters.
#
# Valid Values:
#   true
# Default Values:
#   true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
# Parameter: signingCertDN
#
# Description:
#   Specifies the distinguished name of the root signing certificate that you
#   create when you create the profile. Specify the distinguished name in
#   quotation marks. This default personal certificate is located in the
#   server keystore file. If you do not specifically create or import a root
#   signing certificate, one is created by default. See the
#   -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: signingCertValidityPeriod
#

```

```

# Description:
#   An optional parameter that specifies the amount of time in years that the
#   root signing certificate is valid. If you do not specify this parameter
#   with the -signingCertDN parameter, the root signing certificate is valid
#   for 20 years.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  keyStorePassword
#
# Description:
#   Specifies the password to use on all keystore files created during profile
#   creation. Keystore files are created for the default personal certificate
#   and the root signing certificate.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#

```

```

# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or
#   false. The default value for this parameter is false. When the
#   manageprofiles command-line utility is run with the -enableService option
#   set to true , the Linux service is created with the profile when the
#   command is run by the root user. When a nonroot user runs the
#   manageprofiles command-line utility, the profile is created, but the Linux
#   service is not. The Linux service is not created because the nonroot user
#   does not have sufficient permission to set up the service. An
#   INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#   creation and the profile creation log
#   install_root/logs/manageprofiles/profile_name_create.log contains a
#   message indicating the current user does not have sufficient permission to
#   set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#
# Parameter:  serviceUserName
#
# Description:
#   Specifies the user ID that is used during the creation of the Linux
#   service so that the Linux service will run under this user ID. The Linux
#   service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  winserviceAccountType
#
# Description:
#   The type of the owner account of the Windows service created for the
#   profile. Valid values include specifieduser or localsystem. The
#   localsystem value runs the Windows service under the local account of the
#   user who creates the profile. The default value for this parameter is
#   localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
#   localsystem

```

```

#   specifieduser
# Default Values:
#   localsystem
#
# Parameter: winserviceUserName
#
# Description:
#   Specify your user ID so that the Windows operating system can verify you
#   as an ID that is capable of creating a Windows service. Your user ID must
#   belong to the Administrator group and have the following advanced user
#   rights:
#   Act as part of the operating system
#   Log on as a service
#   The default value for this parameter is the current user name. The value
#   for this parameter must not contain spaces or characters that are not
#   valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#   you specify must have the proper permissions to create a Windows service.
#   You must specify the correct password for the user name that you choose.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winservicePassword
#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID

```

```

#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port
#   number.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
# Parameter:  dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false
#####
# Parameter:  dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:

```

```

#   Varies
# Default Values:
#   CMNDB
#
# Parameter:  dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
# Parameter:  procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.

```



```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPassword
#
# Description:
#   Performance Data Warehouse password. Required for setting up the
#   Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
# Parameter: dbProcSvrMeUserId
#
# Description:
#   Process Server Messaging Engine user name. Required for setting up the
#   Process Server Messaging Engine. If you enter a user-specified a user name
#   and password during profile creation and if you specified ORACLE for
#   dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrMePassword
#
# Description:
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####

```

```

#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMeUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
# The schema name for the Common database. If no value is specified, the
# following default database schema names are used:
# DB2 database
# The default schema name is the value specified by the dbUserId parameter.
# The -cdbSchemaName parameter is valid for all database types except for
# Oracle and Microsoft SQL Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

# Valid Values:
#   Varies
# Default Values:
#   None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: dbAppMeUserId
#
# Description:
#   User name for application messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbAppMePassword
#
# Description:
#   The password for the user id of the dbAppMeUserId parameter.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
# Parameter: dbCeiMeUserId
#
# Description:
#   User name for CEI messaging engine.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbCeiMePassword
#
# Description:
#   The password for the user id of the dbCeiMeUserId parameter. If you enter
#   a user-specified a user name and password during profile creation and if
#   you specified ORACLE for -dbType, this parameter is required.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

```

```

#####
# Parameter: dbSysMeUserId
#
# Description:
#   The default user name for the SCA system bus messaging engine is the first
#   three characters of the Oracle database service name. For example: If the
#   database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbSysMePassword
#
# Description:
#   The default password for the SCA system bus messaging engine is
#   dbPassword.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd
#####
# Parameter: environmentName
#
# Description:
#   Specifies the name of the environment you are configuring. This parameter
#   has no default value, and it must not be empty.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
environmentName=environment_name
#####
# Parameter: environmentType
#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#

```

```

# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter:  isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers.  If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

#####
# Parameter:  webFormConfig
#
# Description:
#   Indicates if Business Space is configured to use IBM Forms Server to work
#   with Human Task Management widgets. The default value for this parameter
#   is false. Indicate true to configure Business Space to use IBM Forms
#   Server. Both the webFormConfig and webFormInstallRoot parameters are
#   required to configure IBM Forms Server. This parameter is valid for
#   stand-alone server profiles only.
#   Note: IBM Forms Server configuration using these parameters is only valid
#   for local IBM Forms Server installations.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter:  webFormHTTP
#
# Description:
#   Specifies the location of the IBM Forms Server Translator. The default URL
#   for the location is http://localhost:8085/translator. No. This parameter
#   is valid only if the -webFormConfig parameter is set to true. It is valid
#   for stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  webFormInstallRoot
#
# Description:
#   Specifies the full path where IBM Forms Server is installed. For example,
#   the IBM Forms Server install root might be

```

```

# C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
# if the -webFormConfig parameter is set to true. It is valid for
# stand-alone server profiles only.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
# Parameter:  configureBRM
#
# Description:
#   Configures the business process rules manager. The default value is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
configureBRM=false
#####
# Parameter:  configureBPC
#
# Description:
#   Determines whether the Business Process Choreographer sample configuration
#   is created.
#   The following conditions apply:
#   Only DB2 databases are supported.
#   The default is set to true for DB2.
#   It uses its own schema within the common database (CMNDB) and the schema
#   name is always BPEDB.
#   Business Process Choreographer database parameters set in the Database
#   Design Tool and generated to the database design file are ignored by
#   Business Process Choreographer at profile creation time.
#   See -bpmdbDesign for information about the database design file.
#   Note: If these conditions are too restrictive, consider setting
#   -configureBPC to false and using other documented methods to configure
#   Business Process Choreographer.
#
# Valid Values:
#   false
#   true
# Default Values:
#   true
#####
configureBPC=true

```

Sample response file for a Standard Process Server stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

# BEGIN COPYRIGHT
# *****
#
# Licensed Materials - Property of IBM
# 5725-C94

```

```

# (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
# This sample program is provided AS IS and may be used, executed, copied
# and modified without royalty payment by customer (a) for its own
# instruction and study, (b) in order to develop applications designed to
# run with an IBM WebSphere product, either for customer's own internal
# use or for redistribution by customer, as part of such an application,
# in customer's own products.
#
# *****
# END COPYRIGHT

#####
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#####

#####
# Parameter: create
#
# Description:
#   Creates the profile.
#   If you are creating a profile, this parameter is required.
#####
create

#####
# Parameter: templatePath
#
# Description:
#   Specifies the directory path to the template files in the installation
#   root directory. Within the profileTemplates directory are various
#   directories that correspond to different profile types and that vary with
#   the type of product installed. The profile directories are the paths that
#   you indicate while using the -templatePath option.
#   Use absolute paths. This parameter must exist as a directory and point to
#   a valid template directory.
#
#   When using the -templatePath parameter, specify the fully qualified file
#   path for the parameter.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
templatePath=BPM/default.procsvr

#####
# Parameter: profileName
#
# Description:
#   Specifies the name of the profile. Use a unique value when creating a
#   profile.
#   Each profile that shares the same set of product binaries must have a
#   unique name. The default profile name is based on the profile type and a
#   trailing number, for example:

```



```

# profileType ProfileNumberwhere profileType is a value such as ProcSrv,
# Dmgr, or Custom and ProfileNumber is a sequential number that creates a
# unique profile name. The value for this parameter must not contain spaces
# or characters that are not valid such as the following: *, ?, ", <, >, ,,
# /, \, and |. The profile name that you choose must not be in use. On
# Windows platforms: If the fully qualified path contains spaces, enclose
# the value in quotation marks. The default value is based on the
# install_root directory, the profiles subdirectory, and the name of the
# file. For example, the default for profile creation is:
# WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
# WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
# file in the install_root/properties directory. The value for this
# parameter must be a valid path for the target system and must not be
# currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
profileName=ProcSrv01

#####
# Parameter: applyPerfTuningSetting
#
# Description:
#   Set a performance tuning level appropriate for the profile you are
#   creating.
#   This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#   development
#   standard
#   production
# Default Values:
#   None
#####
#applyPerfTuningSetting=standard

#####
# Parameter: cellName
#
# Description:
#   Specifies the cell name of the profile. Use a unique cell name for each
#   profile.
#   The default value for this parameter is based on a combination of the
#   short host name, the constant Cell, and a trailing number, for example:
#   if (DMgr)
#     shortHostNameCellCellNumber
#   else
#     shortHostNameNodeNodeNumberCell
#   where CellNumber is a sequential number starting at 01 and NodeNumber is
#   the node number that you used to define the node name.
#   The value for this parameter must not contain spaces or any characters
#   that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#   Varies
# Default Values:
#   Varies
#
# Parameter: nodeName
#
# Description:
#   Specifies the node name for the node that is created with the new profile.
#   Use a unique value within the cell or on the workstation. Each profile
#   that shares the same set of product binaries must have a unique node name.

```

```

# This parameter is required for profile creation only with the
# dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
# procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
# The default value for this parameter is based on the short host name,
# profile type, and a trailing number, for example:
# if (DMgr)
#     shortHostNameCellManagerNodeNumber
# else
#     shortHostNameNodeNodeNumber
# where NodeNumber is a sequential number starting at 01.
# The value for this parameter must not contain spaces or any characters
# that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline. To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:

```

```

# None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
# Parameter: enableAdminSecurity
#
# Description:
# For IBM Business Process Manager omit this parameter. Administrative
# security is always enabled for IBM Business Process Manager profiles.
# You must also specify the parameters -adminUserName and -adminPassword
# along with the values for these parameters.
#
# Valid Values:
# true
# Default Values:
# true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
# Parameter: signingCertDN
#
# Description:
# Specifies the distinguished name of the root signing certificate that you
# create when you create the profile. Specify the distinguished name in
# quotation marks. This default personal certificate is located in the
# server keystore file. If you do not specifically create or import a root
# signing certificate, one is created by default. See the
# -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: signingCertValidityPeriod
#
# Description:
# An optional parameter that specifies the amount of time in years that the
# root signing certificate is valid. If you do not specify this parameter
# with the -signingCertDN parameter, the root signing certificate is valid
# for 20 years.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: keyStorePassword
#
# Description:
# Specifies the password to use on all keystore files created during profile
# creation. Keystore files are created for the default personal certificate
# and the root signing certificate.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period

```

```

#keyStorePassword =keyStore_password

#####
# Parameter:  defaultPorts
#
# Description:
#   Assigns the default or base port values to the profile.d o not use this
#   parameter when using the -startingPort or -portsFile parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  startingPort
#
# Description:
#   Specifies the starting port number for generating and assigning all ports
#   for the profile.
#   Port values are assigned sequentially from the -startingPort value,
#   omitting those ports that are already in use. The system recognizes and
#   resolves ports that are currently in use and determines the port
#   assignments to avoid port conflicts.
#   Do not use this parameter with the -defaultPorts or -portsFile parameters.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter:  portsFile
#
# Description:
#   An optional parameter that specifies the path to a file that defines port
#   settings for the new profile.
#   Do not use this parameter when using the -startingPort or -defaultPorts
#   parameter.
#   During profile creation, the manageprofiles command-line utility uses an
#   automatically generated set of recommended ports if you do not specify the
#   -startingPort parameter, the -defaultPorts parameter or the -portsFile
#   parameter. The recommended port values can be different than the default
#   port values based on the availability of the default ports.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
# Parameter:  enableService
#
# Description:
#   Enables the creation of a Linux service. Valid values include true or

```

```

# false. The default value for this parameter is false. When the
# manageprofiles command-line utility is run with the -enableService option
# set to true , the Linux service is created with the profile when the
# command is run by the root user. When a nonroot user runs the
# manageprofiles command-line utility, the profile is created, but the Linux
# service is not. The Linux service is not created because the nonroot user
# does not have sufficient permission to set up the service. An
# INSTCONPARTIALSUCCESS result is displayed at the end of the profile
# creation and the profile creation log
# install_root/logs/manageprofiles/profile_name_create.log contains a
# message indicating the current user does not have sufficient permission to
# set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
# false
# true
# Default Values:
# None
#
# Parameter: serviceUserName
#
# Description:
# Specifies the user ID that is used during the creation of the Linux
# service so that the Linux service will run under this user ID. The Linux
# service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winserviceAccountType
#
# Description:
# The type of the owner account of the Windows service created for the
# profile. Valid values include specifieduser or localsystem. The
# localsystem value runs the Windows service under the local account of the
# user who creates the profile. The default value for this parameter is
# localsystem. Use this parameter when creating profiles only.
#
# Valid Values:
# localsystem
# specifieduser
# Default Values:
# localsystem
#
# Parameter: winserviceUserName
#
# Description:
# Specify your user ID so that the Windows operating system can verify you
# as an ID that is capable of creating a Windows service. Your user ID must
# belong to the Administrator group and have the following advanced user
# rights:
# Act as part of the operating system
# Log on as a service
# The default value for this parameter is the current user name. The value
# for this parameter must not contain spaces or characters that are not
# valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
# you specify must have the proper permissions to create a Windows service.
# You must specify the correct password for the user name that you choose.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: winservicePassword

```

```

#
# Description:
#   Specify the password for the specified user or the local account that is
#   to own the Windows service.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: winserviceCheck
#
# Description:
#   The value can be either true or false. Specify true to create a Windows
#   service for the server process that is created within the profile. Specify
#   false to not create the Windows service. The default value for this
#   parameter is false.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#
# Parameter: winserviceStartupType
#
# Description:
#   The following values for Windows service startup can be used:
#   manual
#   automatic
#   disabled
#   The default value for this parameter is manual.
#
# Valid Values:
#   manual
#   automatic
#   disabled
# Default Values:
#   manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
# Parameter: bpmdbDesign
#
# Description:
#   Specifies the file that holds the database configuration design for all of
#   the IBM Business Process Manager components .
#   This parameter replaces the -wbldbDesign that was used in previous
#   versions of WebSphere Process Server.
#
#   Note: If a bpmdbDesign file parameter is specified, none of the database
#   related properties should be specified.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
#bpmdbDesign=design_file

```

```

#####
# Parameter: dbCreateNew
#
# Description:
#   Indicates a new database is created or if an existing database is to be
#   reused. Valid values are true or false. This value cannot be true for an
#   Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#   a DB2 database.
#
# Valid Values:
#   false
#   true
# Default Values:
#   None
#####
dbCreateNew=true

#####
# Parameter: dbType
#
# Description:
#   The database type. Set one of the following values for the type of
#   database product you are using with IBM Business Process Manager.
#
#   Required only for one of the following:
#   A stand-alone profile when -bpmdbDesign was not specified.
#   An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   DB2_DATASERVER
# Default Values:
#   DB2_DATASERVER
#
# Parameter: dbJDBCClasspath
#
# Description:
#   The directory path location of JDBC driver files.
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbHostName
#
# Description:
#   The database server host name or IP address. The default value is the
#   local host name. Required when the -bpmdbDesign parameter was not
#   specified. (It is not required if the -bpmdbDesign parameter is
#   specified.)
#
#   Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbServerPort
#
# Description:
#   The database server port number. Depending on the database you are using,
#   you can specify a different port number instead of the default port

```

```

# number.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
# Parameter: dbDelayConfig
#
# Description:
#   Indicates whether to postpone table creation until after the profile is
#   created. Valid values are true or false. The parameter is set to false by
#   default. Setting this parameter to true delays the execution of the
#   scripts that create the database tables. No. Specify false if you want the
#   database to be set up during profile creation.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
dbDelayConfig=false

#####
# Parameter: dbName
#
# Description:
#   The name of the database. By default, the value is set to orcl for Oracle
#   databases and to CMNDB for all other supported databases.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   CMNDB
#
# Parameter: dbUserId
#
# Description:
#   The user ID for all database types. Specifies the user ID that has
#   privileges to create and drop the databases. The WebSphere data source
#   uses this ID to authenticate the database connection.
#   Important: The -dbUserId parameter value must be a valid database
#   authorization ID. Required if the -bpmdbDesign file is not set.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbPassword
#
# Description:
#   The password required for database authentication. Required when the
#   -bpmdbDesign file is not set.

```



```

#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
# Parameter: procSvrDbName
#
# Description:
#   Database name for Process Server database. The value specified for
#   -procSvrDbName and -perfDWDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrUserId
#
# Description:
#   The Process Server user name. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#
# Parameter: dbProcSvrPassword
#
# Description:
#   The Process Server password. Required for setting up the Process Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#   Varies
# Default Values:
#   None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
# Parameter: perfDWDbName
#
# Description:
#   Database name for Performance Data Warehouse. The value specified for
#   -perfDWDbName and -procSvrDbName must be different.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWUserId
#
# Description:
#   Performance Data Warehouse user name. Required for setting up the

```

```

# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter: dbPerfDWPASSWORD
#
# Description:
# Performance Data Warehouse password. Required for setting up the
# Performance Data Warehouse.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
# Parameter: dbProcSvrMeUserId
#
# Description:
# Process Server Messaging Engine user name. Required for setting up the
# Process Server Messaging Engine. If you enter a user-specified a user name
# and password during profile creation and if you specified ORACLE for
# dbType, this parameter is required.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbProcSvrMePassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
# Parameter: dbPerfDWMUserId
#
# Description:
# Performance Data Warehouse Messaging Engine user name. Required for
# setting up the Performance Data Warehouse Messaging Engine.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbPerfDWMPassword
#
# Description:
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:

```

```

# None
#####
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password

#####
# Parameter: cdbSchemaName
#
# Description:
# The schema name for the Common database. If no value is specified, the
# following default database schema names are used:
# DB2 database
# The default schema name is the value specified by the dbUserId parameter.
# The -cdbSchemaName parameter is valid for all database types except for
# Oracle and Microsoft SQL Server.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonUserId
#
# Description:
# The user id to authenticate with the Common database.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#
# Parameter: dbCommonPassword
#
# Description:
# The password for the user id of the dbCommonUserId parameter.
#
# Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#####
# Parameter: environmentName
#
# Description:
# Specifies the name of the environment you are configuring. This parameter
# has no default value, and it must not be empty.
#
# Valid Values:
# Varies
# Default Values:
# None
#####
environmentName=environment_name

#####
# Parameter: environmentType

```

```

#
# Description:
#   Specifies the environment type. The environment type refers to how IBM
#   Business Process Manager is used; for example, in a production, stage or
#   test environment. Load testing might be done on a test server, while a
#   stage environment type might be used as a temporary location to host
#   changes before putting those changes into production. You might specify
#   Stage as the Environment type if the server you are configuring will be
#   accessed and used to review content and new functionality. Valid values
#   are as follows:
#   Test
#   Use Test if the server you are configuring is to be used as a testing
#   environment.
#   Stage
#   Use Stage if the server is to serve as a staging platform to be used as a
#   preproduction server.
#   Production
#   Use Production if the server is to serve in a production capacity.
#   The default value is Test.
#
# Valid Values:
#   Test
#   Production
#   Stage
# Default Values:
#   Test
#####
environmentType=Test

#####
# Parameter: isDeveloperServer
#
# Description:
#   Specifies whether the server is intended for development purposes only.
#   This parameter is useful when creating profiles to test applications on a
#   non-production server prior to deploying the applications on their
#   production application servers. If -isDeveloperServer is set when
#   creating an IBM Business Process Manager profile, then a preconfigured VMM
#   file repository is installed. This file repository contains a sample
#   organization that can be used to test Business Process Choreographer
#   people resolution, ready for you to use as is.
#
# Valid Values:
#   false
#   true
# Default Values:
#   false
#####
#isDeveloperServer=false

```

Augmenting stand-alone profiles for Process Center with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
 - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
 - Run the command `install_root\bin\ProfileManagement\pmt.bat`
 2. Shut down any servers associated with a profile you plan to augment.

3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
 8. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

e. Click **Next**.

10. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 187. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles for Process Server with a DB2 database server using the Profile Management Tool:

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
 - Create a web server definition.
 - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
 - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
 - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
 - Start the tool from the First steps console.
 - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
 - Run the command `install_root\bin\ProfileManagement\pmt.bat`
 2. Shut down any servers associated with a profile you plan to augment.
 3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
 4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
 5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
 6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
 8. On the Process Server Configuration page, set the values for the following parameters.
 - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
 - **Environment Type:** Select how the Process Server is used:
 - Select **Production** if the server is to be used in a production capacity.
 - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
 - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

Restriction: Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this

Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
 - a. From the **Select a database product** list, select **DB2**.
 - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
 - c. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
 - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
 - e. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Specify the following fields.

Table 188. Required database configuration fields for DB2 Database

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/db2/java</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables are all created in the Common database.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

Important: The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

15. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
16. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Augmenting stand-alone profiles using the manageprofiles command-line utility:

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Restriction: To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
 - Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.
1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
 2. Find the appropriate template to use for the augmentation.
Templates for each profile are located in the *install_root*\profileTemplates\BPM directory for BPM templates and under *install_root*\profileTemplates for other products. The following templates are available:
 - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
 - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
 - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
 - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
 - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
 3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir*/profileTemplates/BPM directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC_Std_DMGr_DB2.response
- PC_Std_DMGr_DB2z0S.response
- PC_Std_DMGr_Oracle.response
- PC_Std_DMGr_SQLServer.response
- PC_Adv_DMGr_DB2.response
- PC_Adv_DMGr_DB2z0S.response
- PC_Adv_DMGr_Oracle.response
- PC_Adv_DMGr_SQLServer.response
- PC_Std_Managed_DB2.response
- PC_Std_Managed_DB2z0S.response
- PC_Std_Managed_Oracle.response
- PC_Std_Managed_SQLServer.response
- PC_Adv_Managed_DB2.response
- PC_Adv_Managed_DB2z0S.response
- PC_Adv_Managed_Oracle.response
- PC_Adv_Managed_SQLServer.response
- PC_Std_StandAlone_DB2.response
- PC_Std_StandAlone_DB2z0S.response
- PC_Std_StandAlone_Oracle.response
- PC_Std_StandAlone_SQLServer.response
- PC_Adv_StandAlone_DB2.response
- PC_Adv_StandAlone_DB2z0S.response
- PC_Adv_StandAlone_Oracle.response
- PC_Adv_StandAlone_SQLServer.response
- PS_Std_DMGr_DB2.response
- PS_Std_DMGr_DB2z0S.response
- PS_Std_DMGr_Oracle.response
- PS_Std_DMGr_SQLServer.response
- PS_Adv_DMGr_DB2.response
- PS_Adv_DMGr_DB2z0S.response
- PS_Adv_DMGr_Oracle.response
- PS_Adv_DMGr_SQLServer.response
- PS_Std_Managed_DB2.response
- PS_Std_Managed_DB2z0S.response
- PS_Std_Managed_Oracle.response
- PS_Std_Managed_SQLServer.response
- PS_Adv_Managed_DB2.response
- PS_Adv_Managed_DB2z0S.response
- PS_Adv_Managed_Oracle.response
- PS_Adv_Managed_SQLServer.response
- PS_Std_StandAlone_DB2.response
- PS_Std_StandAlone_DB2z0S.response

- PS_Std_Standalone_Oracle.response
- PS_Std_Standalone_SQLServer.response
- PS_Adv_Standalone_DB2.response
- PS_Adv_Standalone_DB2zOS.response
- PS_Adv_Standalone_Oracle.response
- PS_Adv_Standalone_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv -profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Configuring DB2 databases after stand-alone profile creation:

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating the Process Server database tables:

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root\dbscripts\ProcessServer\DB2* at profile creation time.

2. Run the following script.

```
configProcessServerDB.bat
```

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

Creating the Performance Data Warehouse database tables:

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Chosen not to run the database scripts to initialize the databases during the configuration.
 - Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile_root*\dbscripts\PerformanceDW\DB2\ at profile creation time.
 2. Run the following script.
configPerformanceDWDB.bat

Note: You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

Creating the Common database tables:

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
 - Chosen not to run the database scripts to initialize the Common database during the configuration.
 - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.bat** script. The default directory is *install_root*\profile\dbscripts\CommonDB\DB2\.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db_type* represents the supported database product and *db_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

```
configCommonDB.bat createDB - creates the database and tables
```

configCommonDB.bat - creates only the tables and assumes that the database already exists
Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable_AppScheduler.sql**
- c. **createTable_CommonDB.sql**
- d. **createTable_customization.sql**
- e. **createTable_DirectDeploy.sql**
- f. **createTable_EsbLoggerMediation.sql**
- g. **createTable_governancerepository.sql**
- h. **createTable_lockmanager.sql**
- i. **createTable_mediation.sql**
- j. **createTable_Recovery.sql**
- k. **createTable_RelationshipService.sql**
- l. **insertTable_CommonDB.sql**

Creating the Business Space database tables:

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
 - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.bat** script. The default directory is *install_root\profile\dbscripts\BusinessSpace\node_name_server_name\DB2\CMNDB*.
 2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:
configBusinessSpaceDB.bat Connect to the database, bind the Command Line Interface to the Business Space database using the **db2 bind DB2_installation_directory\bnd\@db2cli.lst blocking all grant public** command, and then reset the database connection.

Loading the database with system information in a stand-alone environment:

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

Important: Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root\profiles/ProcCtr01\bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.bat**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER_INSTALL_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

Configuring a stand-alone environment with a DB2 for z/OS database server on Windows:

You can configure a stand-alone environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

Creating database design files and database scripts (Windows stand-alone):

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS:

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
 - The location of the database tables.
 - The location of the JDBC driver on the system where the server profile will be created.
 - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database

scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
    display help info.

-e db_design_file_name
    edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
    when a db_design_file is given, validation will be done on the specified
    database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
    generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory C:\tmp\DB2-zOS\W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as C:\tmp\DB2-zOS\W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as C:\tmp\DB2-zOS\W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Restriction: If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

Note: Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the `[begin] = BSpace : WBI_BSPACE` section, changing the `databaseName` value to `BSPACE`.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated `BSPACE` for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to `BSPACE` in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the `DbDesignGenerator.sh` command from the `install_root\util\dbUtils` directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- ```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- ```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include `".standalone."`

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

(2)bpm.advanced.standalone.topology, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- ```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
```

```

(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibME] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibME] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]

```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```

[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```

(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server

```

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```

[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :

```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database Location on z/OS system[default=] :
```

```
Database server host[default=] :
```

```
Database server port[default=] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

```
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

```

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]

```

(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]

```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

**Note:** For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM\_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

```
[info] Please pick one of the following [Is this database for a Process Center?(s)] :
```

```

(1>false
(2>true

```

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.standalone**, after you configure the master component, you must manually configure the **BPM\_PerformanceDW** and **BPM\_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

```
[info] Please pick one of the following [database component(s)] :
```

```

(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSpace] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]

```



```
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = complete]
(12)[save and exit]
```

Please enter the number for the database component :12

```
[status] bpm.advanced.standalone.topology is not complete with 6 remaining item(s):
[1] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[2] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'catalog' for VCAT is empty.
[3] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminUserName' for SECURITY_ADMIN_USER is empty.
[4] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property
'adminPassword' for SECURITY_ADMIN_PASSWORD is empty.
[5] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'schemaName' for SCHEMA is empty.
[6] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property
'catalog' for VCAT is empty.
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.standalone is complete with 0 remaining item(s):
```

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment:*

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).



**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commondb
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, *WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB*. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=BPMDB] :

Data source user name[default=] :

Database schema name (SQLID)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :

The user ID you use for administrative security[default=] :

The password for the name specified with the adminUserName parameter[default=] :

Bufferpool of 4K size for tables[default=BP1] :

Bufferpool of 4K size for LOB data[default=BP3] :

Bufferpool of 8K size for tables[default=BP8K1] :

Bufferpool of 16K size for tables[default=BP16K1] :

Bufferpool of 32K size for tables[default=BP32K1] :

Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. To generate database scripts based on the information provided to the database design tool, type *y* and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/ProcessServer for BPM_ProcessServer
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

#### **Related tasks:**

“Creating the messaging engine data stores” on page 2918

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

*Generating scripts for creating database objects in DB2 for z/OS:*

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
  - Database server details
  - Data source details
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.
- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory C:\tmp\DB2-zOS\W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as C:\tmp\DB2-zOS\W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as C:\tmp\DB2-zOS\W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd install_root\util\dbUtils
```

For example: cd C:\IBM\WebSphere\AppServer\util\dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.bat -g db_design_file
```

where *db\_design\_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.bat -g C:\tmp\DB2-zOS\W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

**Tip:** In the database design file, the directories for the SQL scripts are defined by the *ddl\_outDir* variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the *ddl\_outDir* variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.  
To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.
2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

**Note:** The **createDB2.sh** script requires execute permissions.

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

## Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

## Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

## Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating or augmenting stand-alone profiles with a DB2 for z/OS database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Creating Process Center stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does



not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.

- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do



not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths

that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.

- d. Click **Next**.
15. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating Process Server stand-alone profiles with a DB2 for z/OS database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can



include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.

d. Click **Next**.

15. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

16. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.

19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root\bin* directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 

Templates for each profile are located in the *install\_root\profileTemplates\BPM* directory for BPM templates and under *install\_root\profileTemplates* for other products. The following templates are available:

  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root\BPM\samples\manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response



- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

*Sample response file for an Advanced Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_DB2zOS.response
```

```

#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####

```

profileName=ProcCtr01

```

Parameter: applyPerfTuningSetting

Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.

Valid Values:
development
standard
production
Default Values:
None

#applyPerfTuningSetting=standard

Parameter: cellName

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: nodeName

Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: serverName
#
```

```

Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name

```

```

#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:

```

```

Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#

```



```

Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

```

```

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

```

```

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

#####

```

```

Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
```



```

Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```

```

Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is

```

```

localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:

```

```

manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDB0S390
Default Values:
DB2UDB0S390
#####
dbType=DB2UDB0S390

#####
Parameter: dbDelayConfig

```

```

#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:

```



```

Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

```

```

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production

```

```

Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

```

```

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are

```

```

creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

```

```

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#

```

```

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port

```

```

assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:

```



```

Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:

```

```

The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword

```

```

#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing

```

```

environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses DB2z0S.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#

```

```

was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2z0S.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None

```



```

#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName

```

```

#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true

```

```

#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#

```

```

Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:

```

```

false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#

```

```

Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#

```

```

Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId

```



```

#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password
#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or

```

```

test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting stand-alone profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:****Vista****Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. Advanced: Configure the databases using a design file.
    - a. Select **Use a database design file**.
    - b. Click **Browse**.

- c. Specify the fully qualified path name for the design file.  
With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
9. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  10. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  11. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  12. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Augmenting stand-alone profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does

not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.
    - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
    - **Environment Type:** Select how the Process Server is used:
      - Select **Production** if the server is to be used in a production capacity.
      - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
      - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

9. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
10. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
11. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
12. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
13. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.



**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:
  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir*/profileTemplates/BPM directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response

- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv  
 -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```



The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS databases, and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

*Creating and configuring DB2 for z/OS databases after stand-alone profile creation:*

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*DB2 for z/OS authorization prerequisites:*

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

### **User authorization requirements for DB2 for z/OS**

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as `#DB_USER#`. These permissions are available in the `createDatabase.sql` database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the `#DB_USER#` user to create sequences and stored procedures with a schema qualifier of `#SCHEMA#`:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

### **Authorization requirements for views on DB2 for z/OS V10**

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database `#DB_NAME#` can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual

GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;
```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named S1CELL, and connect user WSADMIN to it, as follows:

```
INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
WSADMIN CONNECT 000000 NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE RESUME DATE=NONE
```

### Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.

### **Storage group assignments and buffer pool usage**

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

### **Clustering requirements**

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

#### **Related tasks:**

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full

functionality of IBM Business Process Manager.

*Creating databases in the DB2 for z/OS subsystem:*

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.
- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

### Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

#### Considerations for choosing the **createDB2.sh** script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

#### Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two .sql files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

#### Related tasks:

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

*Configuring the DB2 command line processor:*

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the `createDB2.sh` script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:
  - Modify the `CLASSPATH` environment variable to include the `clp.jar` file.
  - Use the `CLPPROPERTIESFILE` environment variable to define the fully qualified name of the properties file for the command line processor.
  - Define the `db2` command as an alias for the command that starts the command line processor.
  - Specify the `DB2JccConfiguration.properties` file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the `.profile` or `/etc/profile` file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=
/file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=
/wasv8config/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:
  - A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: `server:port/database`. The port is optional, and the DB2 location name must be specified in uppercase characters.
  - A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the `createDB2.sh` script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

**Tip:** When you define a `DB2ALIASNAME` value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSNTIPP ISPf panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the `createDB2.sh` script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two `.sql` files that you can run later using other database tools.

**Related tasks:**



“Creating DB2 for z/OS database objects using the createDB2.sh script”

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2915

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

*Creating DB2 for z/OS database objects using the createDB2.sh script:*

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCcreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**-RunSQL**

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

**Tip:** The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 2911

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

*Creating DB2 for z/OS database objects using the DB2 command line processor:*

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:  

```
cd /u/work/S4CELLDB
```
2. Run the **createDB2.sh** script by using the following syntax:



```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCcreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**Tip:** By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two .sql files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see createDB2.sh script.

The `z_schema.sql` and `z_schemaProc.sql` files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The `z_schemaProc.sql` file contains SQL statements for stored procedures, and is generated in addition to `z_schema.sql` because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the `z_schema.sql` file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql The command line processor reads the data in the file, and
processes the commands in the file consecutively.
```

4. Run the DB2 command line processor again to run the SQL commands in the `z_schemaProc.sql` file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 2911

Before you run the **createdb2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

*Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:*

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

**Tip:** The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

**SPUFI** A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

#### **DSNTEP2**

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:  

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

**Important:** After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

**Tip:** If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:

```
@STOGRP@ = DB2 Storage group name
@DB_NAME@ = DB2 Database name
@SCHEMA@ = DB2 SQLID (Schema Qualifier)
@BP4K@ = Buffer Pool of 4k Size
@BPINDEX@ = Buffer Pool Index
@VOLUMES@ = DB2 comma separated list of volume-ids
@VCAT@ = DB2 integrated catalog facility name
@DB_USER@ = The user ID which will be used to create the database objects
```

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

**Tip:** If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (z\_schema.sql and z\_schemaProc.sql), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

#### Related tasks:

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 2915

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z\_schema.sql and z\_schemaProc.sql) that you can run later by using the DB2 command line processor.

#### Creating the messaging engine data stores:

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.
- When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:

- If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
  - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
    - a. Run the database design tool (**DbDesignGenerator**) from the `install_root/util/dbUtils` directory.
    - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
    - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

**Tip:** As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

**Related tasks:**

“Creating database design files and database scripts for specific components with DB2 for z/OS in a stand-alone environment” on page 2839

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for stand-alone profiles with DB2 for z/OS” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile.

The database design tool also generates database scripts that you can use to create your database tables.

*Granting table privileges to the JCA authentication alias user ID:*

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

**Note:** Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root\profiles/ProcCtr01\bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.bat**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

*Setting the correct schema name for the messaging engines:*

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
  - a. Select **Messaging engines**, and then click the name that is displayed.
  - b. Click **Message store**.
  - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.

- d. Click **Apply**.
  - e. Save your configuration changes.
4. Verify that the messaging engines can be started:
    - a. Log out of the administrative console.
    - b. Stop, and then restart the server.
    - c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:
 

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

*Configuring a stand-alone environment with an Oracle database server on Windows:*

You can configure a stand-alone environment for IBM Business Process Manager using an Oracle database server.

*Setting up Oracle databases:*

You or your database administrator (DBA) can configure the Process Server database, Performance Data Warehouse database, and Common database before, after, or during creating or augmenting a profile.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

You have the following options for creating and configuring the databases:

- You or your DBA creates the databases in advance and the Profile Management Tool configures them for you during profile creation or augmentation.
- You or your DBA creates a database design file using the database design tool, and you pass the file to the Profile Management Tool or **manageprofiles** command-line utility during profile creation or augmentation.
- You postpone the database configuration, and you or your DBA creates the required databases and database tables later by running the scripts that are generated during profile creation or augmentation.

*Creating database design files for Oracle:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for stand-alone profiles with Oracle:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.



- The location of the JDBC driver on the system where the server profile will be created.
- The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the [**begin**] = **BSpace** : **WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- (1)Create a database design for Standalone profile or Deployment Environment
- (2)Create a database design for a single component
- (3)Edit an existing database design
- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2)bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI\_BPC : [status = not complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = not complete]
- (6)[SibME] WBI\_BPC\_ME : [status = not complete]
- (7)[SibME] WBI\_CEI\_ME : [status = not complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12)[save and exit]



4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter *'s'*; or enter anything else to continue :

9. To configure the data source properties component, type anything other than *s* and press Enter. To skip this configuration and accept the defaults, type *s* and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these

components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.advanced.standalone**, you must manually configure the **BPM\_PerformanceDW**, **BPM\_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :3

-----  
[status] BPM\_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB\_USER is empty.

Edit this database component? (y/n) [default=y] :

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

-----  
[status] BPM\_ProcessServer is not complete with 3 remaining item(s):  
[ 1 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'databaseUser' for DB\_USER is empty.  
[ 2 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminUserName' for SECURITY\_ADMIN\_USER is empty.  
[ 3 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminPassword' for SECURITY\_ADMIN\_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\Oracle-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with Oracle in a stand-alone environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm\_performancedw
- (4) bpm\_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 3 to select **(3) Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:



[info] Please enter the values for the properties in the database objects section.  
Database name(SID) [default=BPMDB] :  
Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1>false
- (2>true

Please enter the number for the Is this database for a Process Center? [default=false] :  
The user ID you use for administrative security[default=] :  
The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.  
Database server host[default=] :  
Database server port[default=1521] :

[info] Please pick one of the following [Oracle driver type(s)] :

- (1)oci8
- (2)thin

Please enter the number for the Oracle driver type(s)? [default=thin] :  
Data source user name[default=] :  
Data source password[default=] :  
Oracle JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/Oracle ] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

[info] The script(s) have been generated in

C:\IBM\WebSphere\AppServer\util\dbUtils\Oracle-ProcessServer for BPM\_ProcessServer

[info] thanks, quitting now ...

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

#### *Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

#### **Required property is empty errors**

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
```

```
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
```

```
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

#### **Sample output of running a validation of the existing database design**

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
```

```
...
```

```
[WARNING] 2 potential problems are found in the scripts. They are
```

```
DB_USER @ line 46 in file configCommonDB.bat
```

```
DB_USER @ line 80 in file configCommonDB.sh
```

#### **Contents of the database design tool log file**

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.



*Creating or augmenting stand-alone profiles with an Oracle database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Preparing the Oracle database server:*

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

To configure the database information while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

*Creating Process Center stand-alone profiles with an Oracle database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:**  

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user

or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

### IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
15. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 189. Required database configuration fields for Oracle

| Field                                             | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>Common database</b></p>                     | <p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p>                                                                                                                                                                             |
| <p><b>Process Server database</b></p>             | <p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>                         |
| <p><b>Performance Data Warehouse database</b></p> | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p> |



Table 189. Required database configuration fields for Oracle (continued)

| Field                                              | Action needed                                                                                                                                            |
|----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name.                                                                    |
| Server port                                        | Accept the default value of 1521 or enter the correct server port number.                                                                                |
| Directory location of JDBC driver classpath files  | The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory:<br>\${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle |

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
18. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  19. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

**Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating Process Server stand-alone profiles with an Oracle database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.



The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.

5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. **Advanced:** On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. **Required:** On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.  
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. **Advanced:** On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine



- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 190. Required database configuration fields for Oracle

| Field                                              | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database                                    | <p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p>                                                                                                                                                                             |
| Process Server database                            | <p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>                         |
| Performance Data Warehouse database                | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Server port                                        | Accept the default value of 1521 or enter the correct server port number.                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Directory location of JDBC driver classpath files  | The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory:<br>\${WAS_INSTALL_ROOT}\jdbcdrivers\oracle                                                                                                                                                                                                                                                                                                                                                                       |

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the **manageprofiles** command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root\bin* directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.



1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:
  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail. The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response

- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the

**bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

*Sample response file for an Advanced Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
```

```

a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr.adv
#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01
#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the

```

```

short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####

```

```

Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,

```

```

omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.

```



```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType

```

```

#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.

```

```

An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#

```

```

Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for

```

```

all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: dbAppMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd

#####
Parameter: dbCeiMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#

```



```

Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
user-specified a user name and password during profile creation, this
parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword. If you enter a user-specified a user name and password during
profile creation, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:

```

```

Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#

```

```

Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root
#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false
#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false

```

```
Default Values:
false
#####
configureBPC=false
```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
```

```

path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber

```

```

else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:

```

```

Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.

```



```

Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:

```

```

manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```

```

#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true

```

```

Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId

```

```

#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password
#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password
#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.

```



```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer

```

```

people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that

```

```

you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each

```

```

profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:

```

```

Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the

```

```

root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port

```

```

settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:

```

```

localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID

```



```

#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDdbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID

```

dbProcSvrMePassword=db\_ProcSvr\_Me\_Password

#####

# Parameter: dbPerfDWMeUserId

#

# Description:

# Performance Data Warehouse Messaging Engine user name. Required for  
# setting up the Performance Data Warehouse Messaging Engine. This  
# parameter is required.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbPerfDWMePassword

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbPerfDWMeUserId=db\_PerfDW\_Me\_UserId

dbPerfDWMePassword=db\_PerfDW\_ME\_Password

#####

# Parameter: dbCommonUserId

#

# Description:

# The user id to authenticate with the Common database. For Oracle, the  
# default user name is the first three characters of the Oracle database  
# service name. For example: If the database service name is ORCL,  
# dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business  
# Process Manager profiles and advanced deployment manager (dmgr.\*.adv)  
# profiles, if the -dbType parameter is set to ORACLE or MICROSOFT\_SQLSERVER  
# and no bpmdbDesign file was specified.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#

# Parameter: dbCommonPassword

#

# Description:

# The password for the user id of the dbCommonUserId parameter. Required for  
# all stand-alone IBM Business Process Manager profiles and advanced  
# deployment manager (dmgr.\*.adv) profiles, if the -dbType parameter is set  
# to ORACLE or MICROSOFT\_SQLSERVER and no bpmdbDesign file was specified.

#

# Note: Do not use this parameter if you are using -bpmdbDesign.

#

# Valid Values:

# Varies

# Default Values:

# None

#####

dbCommonUserId=common\_db\_userid

dbCommonPassword=common\_db\_pswd

#####

# Parameter: dbAppMeUserId

#

# Description:

# The default user name is the first three characters of the Oracle database

```

service name. For example: If the database service name is ORCL,
dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd
#####
Parameter: dbCeiMeUserId
#
Description:
The default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
password during profile creation and if you specified ORACLE for dbType,
this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd
#####
Parameter: dbSysMeUserId
#
Description:

```

```

The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
user-specified a user name and password during profile creation, this
parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword. If you enter a user-specified a user name and password during
profile creation, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:

```

```

Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFoMServer/4.0/WebFormServer. No. This parameter is valid only

```



```

if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied

```

```

and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a

```

```

unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.

```

```

procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1
#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####

```

```

processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option

```

```

set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:

```

```

Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew

```



```

#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:

```

```

The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
orcl
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=orc1
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName can be the same.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine. This
parameter is required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

```

```

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage

```

```

Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting stand-alone profiles for Process Center with an Oracle database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:**

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the `runas` command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.

1. Use one of the following methods to start the Profile Management Tool.

- Start the tool from the First steps console.
- Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
- Run the command `install_root\bin\ProfileManagement\pmt.bat`

2. Shut down any servers associated with a profile you plan to augment.

3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
8. Advanced: Configure the databases using a design file.
- a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
- a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine



- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 191. Required database configuration fields for Oracle

| Field                                              | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database                                    | <p>For the Common database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Common database (such as CMNDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p>                                                                                                                                                                             |
| Process Server database                            | <p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>                         |
| Performance Data Warehouse database                | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Server port                                        | Accept the default value of 1521 or enter the correct server port number.                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Directory location of JDBC driver classpath files  | The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory:<br>\${WAS_INSTALL_ROOT}\jdbcdrivers\oracle                                                                                                                                                                                                                                                                                                                                                                       |

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
  - When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
  - The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  13. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### **Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles for Process Server with an Oracle database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.
    - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
    - **Environment Type:** Select how the Process Server is used:
      - Select **Production** if the server is to be used in a production capacity.
      - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
      - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must

run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. You cannot create a new database using Oracle.

**Note:** You must create a separate Oracle user for each messaging engine each time you create a Process Center or Process Server advanced profile. There are five IBM Business Process Manager Advanced messaging engines:

- Process Server Messaging Engine
- Performance Data Warehouse Messaging Engine
- CEI Messaging Engine
- SCA System Messaging Engine
- SCA Application Messaging Engine

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 192. Required database configuration fields for Oracle

| Field           | Action needed                                                                                                                                                                                                                                                                                                                            |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Common database | For the Common database, enter values for the following parameters:<br><br><b>User name</b><br>Enter a user name for the Common database (such as CMNDBA).<br><br><b>Password</b><br>Enter the password that you specified when you created the user in Oracle.<br><br><b>Confirm password</b><br>Confirm the password by reentering it. |

Table 192. Required database configuration fields for Oracle (continued)

| Field                                                     | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Process Server database</b>                            | <p>For the Process Server database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Process Server database (such as BPMDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p>                         |
| <b>Performance Data Warehouse database</b>                | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p><b>User name</b><br/>Enter a user name for the Performance Data Warehouse database (such as PDWDBA).</p> <p><b>Password</b><br/>Enter the password that you specified when you created the user in Oracle.</p> <p><b>Confirm password</b><br/>Confirm the password by reentering it.</p> <p><b>Note:</b> The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p> |
| <b>Database server host name (for example IP address)</b> | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Server port</b>                                        | Accept the default value of 1521 or enter the correct server port number.                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Directory location of JDBC driver classpath files</b>  | The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory:<br>\${WAS_INSTALL_ROOT}\jdbcdrivers\oracle                                                                                                                                                                                                                                                                                                                                                                       |

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
- When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
- The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.



12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.

- default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response



- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

### *Configuring Oracle databases after stand-alone profile creation:*

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

#### *Creating the Process Server database tables:*

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root\dbscripts\ProcessServer\Oracle\* at profile creation time.
2. Run the following script.  
`configProcessServerDB.bat`

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

#### *Creating the Performance Data Warehouse database tables:*

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root\dbscripts\PerformanceDW\Oracle\* at profile creation time.
2. Run the following script.  
`configPerformanceDWDB.bat`

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

### *Creating the Common database tables:*

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
  - Chosen not to run the database scripts to initialize the Common database during the configuration.
  - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.bat** script. The default directory is *install\_root\profile\dbscripts\CommonDB\Oracle\*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db\_type* represents the supported database product and *db\_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run the following command:

**configCommonDB.bat** Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase\_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable\_AppScheduler.sql**
- c. **createTable\_CommonDB.sql**
- d. **createTable\_customization.sql**
- e. **createTable\_DirectDeploy.sql**
- f. **createTable\_EsbLoggerMediation.sql**
- g. **createTable\_governancerepository.sql**
- h. **createTable\_lockmanager.sql**
- i. **createTable\_mediation.sql**
- j. **createTable\_Recovery.sql**
- k. **createTable\_RelationshipService.sql**
- l. **insertTable\_CommonDB.sql**

### *Creating the Business Space database tables:*

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.bat** script. The default directory is *install\_root\profile\dbscripts\BusinessSpace\node\_name\_server\_name\DB2\CMNDB*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:  
**configBusinessSpaceDB.bat**

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root\profiles/ProcCtr01\bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.bat**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

*Configuring a stand-alone environment with an SQL Server database server on Windows:*

You can configure a stand-alone environment for IBM Business Process Manager using a Microsoft SQL Server database server.

*Creating SQL Server databases:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
  - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db\_user\_account** and **db\_user\_password** are the user name and password that are used to log in to create the database, and **database\_name** is the name of the database that you are creating. **COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS** is included because these databases must be case-insensitive.

b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

**Important:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardiisofware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmntExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

*Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts:**javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

**On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

**On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.
- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
- f. Click **OK** again to close the Properties window, and then close Component Services.
- g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

2. Configure the JDBC Distributed Transaction Components:

- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
- b. Extract the archive to any folder.
- c. Copy the sqljdbc\_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc\_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc\_xa.dll file in the x64 folder.
- d. Run the xa\_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc\_xa.dll. These extended stored procedures implement



distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.

- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

#### *Creating database design files for SQL Server:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

#### *Creating database design files for stand-alone profiles with SQL Server:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a stand-alone profile. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.
```

```
-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- ```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- ```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
```



- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the stand-alone profile that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.standalone.”

For example, to configure the database pattern for a stand-alone environment for IBM Business Process Manager Advanced, type the number 2 to select option

**(2)bpm.advanced.standalone.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI\_BPC : [status = not complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = not complete]
- (6)[SibME] WBI\_BPC\_ME : [status = not complete]
- (7)[SibME] WBI\_CEI\_ME : [status = not complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12)[save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for WBI_CommonDB [default=] :
```

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :

Data source password[default=] :

MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (12)[save and exit]

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- 10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.standalone**, after you configure the **BPM\_ProcessServer**, you must manually configure the **BPM\_PerformanceDW** component for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[BPM\_ProcessServer] BPM\_ProcessServer : [master] [status = complete]
- (2)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (3)[SibMe] BPM\_PerformanceDW\_ME : [parent = BPM\_PerformanceDW] [status = complete]
- (4)[SibMe] BPM\_ProcessServer\_ME : [parent = BPM\_ProcessServer] [status = complete]
- (5)[save and exit]

```
Please enter the number for the database component :2
```

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

- 11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.standalone is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with SQL Server in a stand-alone environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.

[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

- (1)Create a database design for Standalone profile or Deployment Environment
- (2)Create a database design for a single component
- (3)Edit an existing database design
- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm\_performancedw
- (4)bpm\_processserver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Process Server database, you see a series of prompts similar to the following example:



[info] Please enter the values for the properties in the database objects section.  
Database name[default=BPMDDB] :  
Database server host[default=] :  
Database User name[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1)false  
(2>true

Please enter the number for the Is this database for a Process Center? [default=false] :  
The user ID you use for administrative security[default=] :  
The password for the name specified with the adminUserName parameter[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.  
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

(1) false  
(2) true

Data source user name[default=] :  
Data source password[default=] :  
MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.standalone.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\SQLServer-ProcessServer for BPM_ProcessServer
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.



*Creating or augmenting stand-alone profiles with an SQL Server database server:*

You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility. You can augment an existing profile instead of creating a new one if you have an existing WebSphere Application Server V8.0 profile.

*Preparing the SQL Server database server:*

During the profile creation or augmentation process, you can choose to configure the Process Server database, Performance Data Warehouse database, and Common database. Alternatively, you can configure the databases using a design file that was generated by the database design tool.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

**Restriction:** The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

*Table 193.*

| Database management system | Minimum privilege required to use the data store tables                                                                                                                                                                                        | Additional privilege required to create the data store tables |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| Microsoft SQL Server       | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege.    |

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ\_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the CMNDB database as owner dbo using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDBE00 (Business Process Choreographer Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)

- MEDPE00 (Performance Data Warehouse messaging engine)

IBM Business Process Manager packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the Detailed hardware and software requirements for IBM Business Process Manager page.

**Note:** You are responsible for providing JDBC driver levels outside of what is packaged with IBM Business Process Manager.

You must configure XA transactions after the database is installed and before you start the server.

*Creating Process Center stand-alone profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Center using the Profile Management Tool.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. The advanced option is recommended for SQL Server. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.

- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Center configuration and expand the section. Select the **IBM BPM Advanced, Process Center Stand-alone** profile and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you selected the Windows Authentication option, you must change the default **LocalSystem** ID to the current user ID and password (the user that launched the Profile Management Tool) so that the Windows service will be created correctly.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.



15. Advanced: Configure the databases using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

16. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
17. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 194. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Action needed                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.</p> <p><b>Restriction:</b></p> <ul style="list-style-type: none"> <li>• If you are using <b>Typical</b> profile creation, you must update the Windows service that has been created for you. Change it from <b>LocalSystem</b> to the specific user ID that you used when creating the profile.</li> <li>• If you are using <b>Advanced</b> profile creation (recommended), make sure that on the Windows Service Definition page, you change the default <b>LocalSystem</b> ID to the current user ID and password (the user that launched the Profile Management Tool) so that the Windows service will be created correctly.</li> </ul> |                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Common database</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | <p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Common database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Common database.</li> </ul> |

Table 194. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

| Field                                              | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Process Server database                            | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Process Server database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Process Server database.</li> </ul>                                                 |
| Performance Data Warehouse database                | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                                                                                                                                                                                             |
| Server port                                        | Accept the default value of 1433 or enter the correct server port number.                                                                                                                                                                                                                                                                                                                                                                                         |
| Directory location of JDBC driver classpath files  | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer                                                                                                                                                                                                                                                                                           |

Table 195. Required database configuration fields for Microsoft SQL Server - Part 3

| Field                                                                                                                                                                                                                                                                                                                             | Action required for SQL Authentication                                                                                                                                                                                                                   | Action required for Windows Authentication                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul> | <p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p> | <p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p> |

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- **SQL Authentication:** The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
  - **Windows Authentication:** With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.
18. **Advanced:** On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
  19. **Advanced:** Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
  20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
  21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
    - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
    - Optionally, configure Business Process Choreographer.
    - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### **Related tasks:**

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating Process Server stand-alone profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a stand-alone profile for Process Server using the Profile Management Tool.



If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. The advanced option is recommended for SQL Server. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Advanced: Process Server configuration and expand the section. Select the **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**.
5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended) and the default WebSphere Application Server application. Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.  
Click **Next**. If you selected **Typical** profile creation, skip to the Process Server Configuration step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate

- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. **Advanced:** On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.securi ty` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you selected the Windows Authentication option, you must change the default **LocalSystem** ID to the current user ID and password (the user that launched the Profile Management Tool) so that the Windows service will be created correctly.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Optional: Advanced: To include a web server definition in the profile, perform the following steps:
  - a. Select **Create a web server definition**.
  - b. Specify the web server characteristics on the page, and click **Next**
  - c. Specify the web server characteristics on Part 2 of the page. If you use a web server to route requests to IBM Business Process Manager, you must include a web server definition. You can include the definition now, or define the web server to IBM Business Process Manager later. If you include the web server definition during the creation of this profile, you can install the web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the web server after you create this profile, you must define the web server in a separate profile.
  - d. Click **Next**.
15. On the Process Server Configuration page, set the values for the following parameters.
  - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.
  - **Environment Type:** Select how the Process Server is used:
    - Select **Production** if the server is to be used in a production capacity.
    - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
    - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when

deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

16. Advanced: Configure the databases using a design file.

- a. Select **Use a database design file**.
- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.
- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify your database information.

- a. From the **Select a database product** list, select **Microsoft SQL Server**.
- b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
- c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
- d. Click **Next**.

18. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.



Table 196. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.</p> <p><b>Restriction:</b></p> <ul style="list-style-type: none"> <li>• If you are using <b>Typical</b> profile creation, you must update the Windows service that has been created for you. Change it from <b>LocalSystem</b> to the specific user ID that you used when creating the profile.</li> <li>• If you are using <b>Advanced</b> profile creation (recommended), make sure that on the Windows Service Definition page, you change the default <b>LocalSystem</b> ID to the current user ID and password (the user that launched the Profile Management Tool) so that the Windows service will be created correctly.</li> </ul> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <p><b>Common database</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | <p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Common database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Common database.</li> </ul>                                                                                 |
| <p><b>Process Server database</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | <p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Process Server database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Process Server database.</li> </ul>                                                 |
| <p><b>Performance Data Warehouse database</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul> |
| <p><b>Database server host name (for example IP address)</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <p>Accept the default value of localhost or enter the correct database server host name.</p>                                                                                                                                                                                                                                                                                                                                                                             |
| <p><b>Server port</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <p>Accept the default value of 1433 or enter the correct server port number.</p>                                                                                                                                                                                                                                                                                                                                                                                         |
| <p><b>Directory location of JDBC driver classpath files</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | <p>The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</p>                                                                                                                                                                                                                                                                                           |

Table 197. Required database configuration fields for Microsoft SQL Server - Part 3

| Field                                                                                                                                                                                                                                                                                                                             | Action required for SQL Authentication                                                                                                                                                                                                                   | Action required for Windows Authentication                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul> | <p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p> | <p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p> |

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
  - Optionally, configure Business Process Choreographer.
  - Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Creating stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root\bin* directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).
 

Templates for each profile are located in the *install\_root\profileTemplates\BPM* directory for BPM templates and under *install\_root\profileTemplates* for other products. The following templates are available:

  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.



The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2z0S.response

- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

#### 4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating stand-alone profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

**Important:** If you are setting **-dbType MSSQLSERVER\_MICROSOFT**, **-dbWinAuth true**, and **-winserviceCheck true**, you must also set **-winserviceAccountType specifieduser**, **-winserviceUserName dbUser**, and **-winservicePassword dbPassword**.

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a stand-alone profile that includes the CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space components in one database.

```
install_root\bin\manageprofiles -create -templatePath install_root/profileTemplates/BPM/
default.procsvr.adv -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath
path to JDBC driver -dbUserId userID -dbPassword password -dbCommonForME true -dbCeMeSchemaName
ceiSchema -dbSysMeSchemaName sysSchema
-dbAppMeSchemaName appSchema
```

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Response file for creating stand-alone profiles:*

You can use the response file provided here with the **manageprofiles** command to create a stand-alone profile.

Sample response file for an Advanced Process Center stand-alone profile:

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
```

```

#####
templatePath=BPM/default.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:

```



```

The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:

```

```

manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#

```

```

Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the

```

```

-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password
#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:

```

```

Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserId
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

```

```

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: dbAppMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies

```

```

Default Values:
None
#####
dbAppMeSchemaName=app_me_chemaName
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

#####
Parameter: dbCeiMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#

```



```

Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeSchemaName=sys_me_schemaName
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.

```

```

Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false
#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work
with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot

```

```

#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.
The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Center stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procctr

#####
Parameter: profileName
#
Description:

```

```

Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcCtr01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName

```

```

#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod

```

```

#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile

```

```

#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:

```



```

localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true

```

```

#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#

```

```

Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPassword
#

```

```

Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database

```

```

service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmbdDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmbdDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmbdDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmbdDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmbdDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name
#####
Parameter: environmentType
#
Description:

```

```

Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,

```

```

in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose

```



```

the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None
#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)

```

```

shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

```

```

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:

```

```

Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user

```

```

does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an

```

```

Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#

```



```

Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDbName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDbName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####

```

```

Parameter: dbAppMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMeUserId
#
Description:
User name for application messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbAppMePassword
#
Description:
The password for the user id of the dbAppMeUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbAppMeSchemaName=app_me_chemaName
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd
#####
Parameter: dbCeiMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMeUserId
#
Description:
User name for CEI messaging engine.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCeiMePassword
#
Description:
The password for the user id of the dbCeiMeUserId parameter. If you enter
a user-specified a user name and password during profile creation and if
you specified ORACLE for -dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
dbCeiMeSchemaName=cei_me_chemaName
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd

#####
Parameter: dbSysMeSchemaName
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMeUserId
#
Description:
The default user name for the SCA system bus messaging engine is the first
three characters of the Oracle database service name. For example: If the
database service name is ORCL, dbSysMeUserId is ORCSS00.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbSysMePassword
#
Description:
The default password for the SCA system bus messaging engine is
dbPassword.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbSysMeSchemaName=sys_me_schemaName
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentName
#
Description:

```

```

Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

#####
Parameter: webFormConfig
#
Description:
Indicates if Business Space is configured to use IBM Forms Server to work

```

```

with Human Task Management widgets. The default value for this parameter
is false. Indicate true to configure Business Space to use IBM Forms
Server. Both the webFormConfig and webFormInstallRoot parameters are
required to configure IBM Forms Server. This parameter is valid for
stand-alone server profiles only.
Note: IBM Forms Server configuration using these parameters is only valid
for local IBM Forms Server installations.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: webFormHTTP
#
Description:
Specifies the location of the IBM Forms Server Translator. The default URL
for the location is http://localhost:8085/translator. No. This parameter
is valid only if the -webFormConfig parameter is set to true. It is valid
for stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: webFormInstallRoot
#
Description:
Specifies the full path where IBM Forms Server is installed. For example,
the IBM Forms Server install root might be
C:/IBM/IBMFoMSServer/4.0/WebFormServer. No. This parameter is valid only
if the -webFormConfig parameter is set to true. It is valid for
stand-alone server profiles only.
#
Valid Values:
Varies
Default Values:
None
#####
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

#####
Parameter: configureBRM
#
Description:
Configures the business process rules manager. The default value is false.
#
Valid Values:
false
true
Default Values:
false
#####
configureBRM=false

#####
Parameter: configureBPC
#
Description:
Determines whether the Business Process Choreographer sample configuration
is created.
The following conditions apply:
Only DB2 databases are supported.

```

```

The default is set to true for DB2.
It uses its own schema within the common database (CMNDB) and the schema
name is always BPEDB.
Business Process Choreographer database parameters set in the Database
Design Tool and generated to the database design file are ignored by
Business Process Choreographer at profile creation time.
See -bpmdbDesign for information about the database design file.
Note: If these conditions are too restrictive, consider setting
-configureBPC to false and using other documented methods to configure
Business Process Choreographer.
#
Valid Values:
false
Default Values:
false
#####
configureBPC=false

```

*Sample response file for a Standard Process Server stand-alone profile:*

Modify and use this response file to create a stand-alone profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Standalone PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Standalone_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

```

```

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/default.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=ProcSrv01

#####
Parameter: applyPerfTuningSetting
#
Description:
Set a performance tuning level appropriate for the profile you are
creating.
This parameter is a WebSphere Application Server parameter.
#
Valid Values:
development
standard
production
Default Values:
None

```



```

#####
#applyPerfTuningSetting=standard

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: serverName
#
Description:
Specifies the name of the server.
#####
cellName=cell_name
nodeName=node_name
serverName=server1

#####
Parameter: processCenterURL
#
Description:
Specifies the URL of the Process Center that the Process Server connects
to. If this parameter is set, the protocol must be either http or https.
Note: No corresponding useServerOffline parameter exists to use the server
offline. To use the server offline, omit the processCenterURL parameter.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterUserId
#
Description:
Specifies the user ID to be used to connect to the Process Center. This
parameter is valid for stand-alone Process Server profiles. This parameter
is required if the Process Server is online (connected to a Process
Center). It is required for a stand-alone process server (standard or
advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: processCenterPassword
#
Description:
Specifies the user password to be used to connect to the Process Center.
This parameter is valid for stand-alone Process Server profiles. This
parameter is required if the Process Server is online (connected to a
Process Center). It is required for a stand-alone process server (standard
or advanced) profile if the processCenterURL parameter is specified.
#
Valid Values:
Varies
Default Values:
None
#####
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:

```

```

Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The

```

```

localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic

```

```

disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCclasspath
#

```

```

Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: procSvrDbName
#
Description:
Database name for Process Server database. The value specified for
-procSvrDbName and -perfDWDName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrUserId
#
Description:
The Process Server user name. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrPassword
#
Description:
The Process Server password. Required for setting up the Process Server.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

#####
Parameter: perfDWDName
#
Description:
Database name for Performance Data Warehouse. The value specified for
-perfDWDName and -procSvrDbName must be different.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWUserId
#
Description:
Performance Data Warehouse user name. Required for setting up the
Performance Data Warehouse.

```



```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Parameter: dbPerfDWPASSWORD
#
Description:
Performance Data Warehouse password. Required for setting up the
Performance Data Warehouse.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#####
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPASSWORD=db_PerfDW_Password

#####
Parameter: dbProcSvrMeUserId
#
Description:
Process Server Messaging Engine user name. Required for setting up the
Process Server Messaging Engine. If you enter a user-specified a user name
and password during profile creation and if you specified ORACLE for
dbType, this parameter is required.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbProcSvrMePassword
#
Description:
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

#####
Parameter: dbPerfDWMeUserId
#
Description:
Performance Data Warehouse Messaging Engine user name. Required for
setting up the Performance Data Warehouse Messaging Engine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPerfDWMePassword
#
Valid Values:
Varies
Default Values:
None
#####
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password

#####

```

```

Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database. For Oracle, the
default user name is the first three characters of the Oracle database
service name. For example: If the database service name is ORCL,
dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
Process Manager profiles and advanced deployment manager (dmgr.*.adv)
profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter. Required for
all stand-alone IBM Business Process Manager profiles and advanced
deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true
#####
Parameter: environmentName
#
Description:
Specifies the name of the environment you are configuring. This parameter
has no default value, and it must not be empty.
#
Valid Values:
Varies
Default Values:
None
#####
environmentName=environment_name

```

```

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting stand-alone profiles for Process Center with an SQL Server database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Center using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  8. Advanced: Configure the databases using a design file.
    - a. Select **Use a database design file**.
    - b. Click **Browse**.
    - c. Specify the fully qualified path name for the design file.
    - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

9. On the Database Configuration - Part 1 page, specify your database information.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the `bootstrapProcessServerData` command.
  - d. Click **Next**.
10. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 198. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                        | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>If you select the <b>Apply Windows Authentication</b> option, and you have created a Windows service during profile creation, make sure the Windows service is configured to log on with the user ID used for Windows authentication.</p> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <p><b>Common database</b></p>                                                                                                                                                                                                                | <p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Common database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Common database.</li> </ul>                                                                                 |
| <p><b>Process Server database</b></p>                                                                                                                                                                                                        | <p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Process Server database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Process Server database.</li> </ul>                                                 |
| <p><b>Performance Data Warehouse database</b></p>                                                                                                                                                                                            | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul> |
| <p><b>Database server host name (for example IP address)</b></p>                                                                                                                                                                             | <p>Accept the default value of localhost or enter the correct database server host name.</p>                                                                                                                                                                                                                                                                                                                                                                             |
| <p><b>Server port</b></p>                                                                                                                                                                                                                    | <p>Accept the default value of 1433 or enter the correct server port number.</p>                                                                                                                                                                                                                                                                                                                                                                                         |
| <p><b>Directory location of JDBC driver classpath files</b></p>                                                                                                                                                                              | <p>The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code></p>                                                                                                                                                                                                                                                                              |

Table 199. Required database configuration fields for Microsoft SQL Server - Part 3

| Field                                                                                                                                                                                                                                                                                                                             | Action required for SQL Authentication                                                                                                                                                                                                                   | Action required for Windows Authentication                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul> | <p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p> | <p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p> |

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

11. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.



12. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
13. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
14. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles for Process Server with an SQL Server database server using the Profile Management Tool:*

If you have an existing WebSphere Application Server V8.0 stand-alone profile, you can configure a stand-alone profile for Process Server using the Profile Management Tool.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure required databases while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

**Vista**

**Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the **pmt.bat** file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a



com.ibm.wsspi.profile.WSProfileException or Access is denied message occurs in the *install\_root\bin\ProfileManagement\pmt.bat* file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command *install\_root\bin\ProfileManagement\pmt.bat*
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. Click **Next**. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
  8. On the Process Server Configuration page, set the values for the following parameters.
    - **Environment name:** The environment name is used to connect from a Process Center to this Process Server.

- Environment **Type**: Select how the Process Server is used:
  - Select **Production** if the server is to be used in a production capacity.
  - Select **Stage** if the server is to be used as a temporary location to host changes before putting them into production.
  - Select **Test** if the server is to be used as a testing environment, for example, for load testing.

**Restriction:** Do not mix production and non-production servers in the same cell.

Select **Use this server offline** if you did not select **Production** as the environment type or if this Process Server will not be connected to a Process Center. Offline servers can still be used when deploying snapshots of process applications, but the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

If you did not select **Use this server offline**, provide the following information for the Process Center that this server is to connect to:

- **Protocol:** Select either **http://** or **https://** as the connection protocol to the Process Center.
- **Host name:** Enter the host or virtual host that this Process Server will use to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.
- **Port:** Enter the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.
- **User name:** Enter the name of a Process Center user. Process Server will connect to Process Center as this user.
- **Password:** Enter the password for the Process Center user.

You can click **Test Connection** to check the connection to the Process Center.

If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

9. Advanced: Configure the databases using a design file.

- a. Select **Use a database design file**.
- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.
- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

10. On the Database Configuration - Part 1 page, specify your database information.

- a. From the **Select a database product** list, select **Microsoft SQL Server**.
- b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
- c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

d. Click **Next**.

11. On the Database Configuration - Part 2 page, complete the database configuration. Then click **Next** to go to the Database Configuration - Part 3 page. Specify the following fields.

Table 200. Required database configuration fields for Microsoft SQL Server - Part 2

| Field                                                                                                                                                                                                                                 | Action needed                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| If you select the <b>Apply Windows Authentication</b> option, and you have created a Windows service during profile creation, make sure the Windows service is configured to log on with the user ID used for Windows authentication. |                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Common database</b>                                                                                                                                                                                                                | For the Common database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Common database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Common database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Common database.</li> </ul>                                                                                 |
| <b>Process Server database</b>                                                                                                                                                                                                        | For the Process Server database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Process Server database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Process Server database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Process Server database.</li> </ul>                                                 |
| <b>Performance Data Warehouse database</b>                                                                                                                                                                                            | For the Performance Data Warehouse database, enter values for the following parameters: <ul style="list-style-type: none"> <li>• <b>User name</b><br/>Enter the Performance Data Warehouse database user name.</li> <li>• <b>Password</b><br/>Enter a password to authenticate with the Performance Data Warehouse database.</li> <li>• <b>Confirm password</b><br/>Confirm the password to authenticate with the Performance Data Warehouse database.</li> </ul> |
| <b>Database server host name (for example IP address)</b>                                                                                                                                                                             | Accept the default value of localhost or enter the correct database server host name.                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Server port</b>                                                                                                                                                                                                                    | Accept the default value of 1433 or enter the correct server port number.                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>Directory location of JDBC driver classpath files</b>                                                                                                                                                                              | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer                                                                                                                                                                                                                                                                                           |

Table 201. Required database configuration fields for Microsoft SQL Server - Part 3

| Field                                                                                                                                                                                                                                                                                                                             | Action required for SQL Authentication                                                                                                                                                                                                                   | Action required for Windows Authentication                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• CEI messaging engine schema name</li> <li>• SCA Application bus messaging engine schema name</li> <li>• SCA system bus messaging engine schema name</li> <li>• Process Server messaging engine schema name</li> <li>• Performance Data Warehouse messaging engine schema name</li> </ul> | <p>Messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.</p> <p><b>Note:</b> For SQL Authentication, the schema names are mapped to the user IDs.</p> | <p>Messaging engine tables must be configured with different schema names. They cannot use the same schema name as any other messaging engine.</p> |

If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a stand-alone profile, and put the messaging engine tables into the Common database, you must create the following schemas in the **CMNDB** database as owner **dbo** using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

**Important:** You must configure these schemas either before you start the server and messaging engine (recommended) or while the messaging engine is starting. Note that these are the default values if Windows Authentication is selected. You can specify different values. For SQL Authentication, the values are the same as the specified user IDs.

- MEDSS00 (Service Component Architecture (SCA) System Bus Messaging Engine)
- MEDSA00 (Service Component Architecture (SCA) Application Bus Messaging Engine)
- MEDCM00 (Common Event Infrastructure Bus Messaging Engine)
- MEDPS00 (Process Server messaging engine)
- MEDPE00 (Performance Data Warehouse messaging engine)

For stand-alone configurations only, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database. The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables are all created in the Common database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- SQL Authentication: The Service Component Architecture (SCA) system, SCA application, and Common Event Infrastructure (CEI) messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.
- Windows Authentication: With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

12. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

13. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

You have augmented a WebSphere Application Server V8.0 profile with an IBM Business Process Manager profile.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Augmenting stand-alone profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a stand-alone profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
 Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:
  - default.procctr: for a Process Center stand-alone profile for IBM BPM Standard or Express.
  - default.procctr.adv: for a Process Center stand-alone profile for IBM BPM Advanced.
  - default.procsvr: for a Process Server stand-alone profile for IBM BPM Standard or Express.
  - default.procsvr.adv: for a Process Server stand-alone profile for IBM BPM Advanced.
  - default.esbserver: for a WebSphere Enterprise Service Bus stand-alone server profile.

3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2z0S.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`
- `PS_Adv_DMGr_DB2.response`
- `PS_Adv_DMGr_DB2z0S.response`



- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/default.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- If you did not run the database scripts during profile creation, run the scripts to create the databases and load the Process Server database with system information using the **bootstrapProcessServerData** command.
- Optionally, configure Business Process Choreographer.
- Proceed to the First Steps Console and start the server for the stand-alone profile you created.

#### Related tasks:

Loading the database with system information in a stand-alone environment

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

*Configuring SQL Server databases after stand-alone profile creation:*

If you did not create the tables for the databases while creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

### *Creating the Process Server database tables:*

If you postponed creating the tables for the Process Server database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone server profile.
- Chosen not to run the database scripts to initialize the databases during the configuration.
- Created the database that you want to create the Process Server database tables in. The default name for the Process Server database is **BPMDB**.

Do not perform these steps if you plan to create a network deployment environment.

1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root*\dbscripts\ProcessServer\SQLServer\ at profile creation time.
2. Run the following script.  
configProcessServerDB.bat

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Process Server and have completed the database configuration.

You can now load the database with system information and start the server.

### *Creating the Performance Data Warehouse database tables:*

If you postponed creating the database tables for the IBM Performance Data Warehouse until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Chosen not to run the database scripts to initialize the databases during the configuration.
  - Created the database that you want to create the Performance Data Warehouse database tables in. The default name for the Performance Data Warehouse database is **PDWDB**.
1. Go to the directory that holds the SQL scripts for creating the database tables. By default, SQL scripts are output to *profile\_root*\dbscripts\PerformanceDW\SQLServer\ at profile creation time.
  2. Run the following script.  
configPerformanceDWDB.bat

**Note:** You can specify the **createDB** parameter after the command to create a local database. Skip this parameter if you have already created the required database.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

### *Creating the Common database tables:*

If you postponed creating the Common database tables until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.



You have performed the following actions:

- Created or augmented a stand-alone server profile.
  - Chosen not to run the database scripts to initialize the Common database during the configuration.
  - Created the database that you want to create the Common database tables in.
1. Go to the directory containing the **configCommonDB.bat** script. The default directory is *install\_root\profile\dbscripts\CommonDB\SQLServer\*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

The variable *db\_type* represents the supported database product and *db\_name* represents the name of the database.

You must pass the **createDB** parameter to the script if you want to create a new local database; otherwise an existing database will be used. Run one of the following commands:

**configCommonDB.bat createDB** - creates the database and tables

**configCommonDB.bat** - creates only the tables and assumes that the database already exists

Alternatively, you could customize and run the individual scripts manually in the following order:

- a. Run **createDatabase\_CommonDB.sql** if the Business Space database and the CommonDB are not the same database instance, and then connect to the database created.
- b. **createTable\_AppScheduler.sql**
- c. **createTable\_CommonDB.sql**
- d. **createTable\_customization.sql**
- e. **createTable\_DirectDeploy.sql**
- f. **createTable\_EsbLoggerMediation.sql**
- g. **createTable\_governancerepository.sql**
- h. **createTable\_lockmanager.sql**
- i. **createTable\_mediation.sql**
- j. **createTable\_Recovery.sql**
- k. **createTable\_RelationshipService.sql**
- l. **insertTable\_CommonDB.sql**

*Creating the Business Space database tables:*

If you postponed creating the Business Space database tables in the Common database until after profile creation, you or your database administrator must create the tables manually. You can use the scripts that the Profile Management Tool generates during profile creation or augmentation.

You have performed the following actions:

- Created or augmented a stand-alone profile.
  - Created the database that you want to create the Business Space database tables in. The default name for the Common database is **CMNDB**.
1. Go to the directory containing the **configBusinessSpaceDB.bat** script. The default directory is *install\_root\profile\dbscripts\BusinessSpace\node\_name\_server\_name\DB2\CMNDB*.
  2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. Run the following command:

**configBusinessSpaceDB.bat**

*Loading the database with system information in a stand-alone environment:*

If you created a stand-alone profile and postponed creating the database tables, you must run the **bootstrapProcessServerData** command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly.

**Note:** This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation or augmentation.

If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone profile directory. For example:

```
install_root\profiles\ProcCtr01\bin
```

Run the bootstrap utility using the following syntax:

- **bootstrapProcessServerData.bat**

For a stand-alone profile, the **-serverName** and **-nodeName** parameters may be omitted; the **bootstrapProcessServerData** command automatically finds the one and only server in the stand-alone profile.

You have loaded the database with system information before successfully starting the Process Server or Process Center. The log information for the bootstrap operation is saved to the *USER\_INSTALL\_ROOT/logs/* directory in a file called *bootstrapProcessServerData.timestamp.log*. The console displays a subset of the logged information.

**Important:** If the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
```

```
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData.sh
```

### **Modifying an existing installation:**

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

## **Configuring a network deployment environment on Windows**

For a network deployment configuration, you install the software and then create or augment a deployment manager, create or augment at least one custom profile, and configure the network deployment environment.

### **Granting write permission of files and directories to nonadministrative users for profile creation or augmentation:**

If you are not the user who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. The product installer can grant this permission or create a group with permission to create or augment profiles.

The product installer (who can be an administrative or nonadministrative user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonadministrative users. The nonadministrative users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

Nonadministrative users create their own profiles to manage their own environments. Typically, they manage environments for development purposes.

Nonadministrative users must store their profiles in their private directory structure, not in the *install\_root*\profiles directory of the product.

### **Restrictions:**

- IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonadministrative users. A nonadministrative user cannot augment profiles owned by another user.
- Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonadministrative users. The nonadministrative user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonadministrative users a range of values for each of the fields, and assign responsibility to the users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, certain directories and files were created. You can skip the steps in this topic that create these directories and files. If no profile was previously created, you must complete the steps to create the required directories and files.

The following example task shows how to create a group that is authorized to create profiles. The terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager. The installer can perform the following steps to create the **profilers** group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. The product installer can be a administrative or nonadministrative user.
2. Using operating system commands, perform the following steps:
  - a. Create a group named **profilers**, which will contain all users who can create profiles.
  - b. Create a user named **user1**, who can create profiles.
  - c. Add users **product\_installer** and **user1** to the **profilers** group.
3. Log off and log back on as the installer to pick up the new group.
4. If no profile exists, create the following directories as the installer:
  - Create the *install\_root*\logs\manageprofiles directory by following instructions in the Windows documentation. For this example procedure, the directory is:  
*install\_root*\logs\manageprofiles
  - Create the *install\_root*\properties\fsdb directory by following instructions in the Windows documentation. For this example procedure, the directory is:  
*install\_root*\properties\fsdb
5. If no profile exists, create the profileRegistry.xml file as the installer. For this example, the file path is:  
*install\_root*\properties\profileRegistry.xml

Add the following information to the profileRegistry.xml file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions. The following example assumes that the variable **\$WASHOME** is the IBM Business Process Manager root installation directory **C:\Program Files\IBM\WebSphere\AppServer**. Follow instructions in the Windows documentation to give the **profilers** group read and write permission to the following directories and their files:  
@WASHOME\logs\manageprofiles  
@WASHOME\properties  
@WASHOME\properties\fsdb  
@WASHOME\properties\profileRegistry.xml

You might have to change the permissions on additional files if the nonadministrative user encounters permission errors. For example, if the product installer authorizes a nonadministrative user to delete a profile, then the product installer might have to delete the following file:

```
install_root\properties\profileRegistry.xml_LOCK
```

Give write access to the nonadministrative user for the file to authorize the user to delete the file. If the nonadministrative user still cannot delete the profile, then the product installer can delete the profile.

The installer created the **profilers** group and gave the group the correct permissions to the directories and files required for a nonadministrative user to create profiles.

The nonadministrative user that belongs to the **profilers** group can create profiles in a directory that the nonadministrative user owns and to which the nonadministrative user has write permission. However, the nonadministrative user cannot create profiles in the installation root directory of the product.

The administrative user and the nonadministrative user can use the same tasks to manage profiles.

## Creating databases and database design files:

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. You must create the databases before network deployment.

### *Creating databases and database design files for DB2:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

### *Creating DB2 databases:*

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

Use the following command to create the databases for DB2 (replace the @ placeholder strings with the appropriate values):

```
create database @DB_NAME@ automatic storage yes using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

### *Creating database design files for DB2:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Creating database design files for deployment environments with DB2:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.

- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```



**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include ".nd."

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than *s* and press Enter. To skip this configuration and accept the defaults, type *s* and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:



[info] Please pick one of the following [database provider(s)] :

(1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

-----

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the master component, all database components are listed as complete.

[info] Please pick one of the following [database component(s)] :

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [status = complete]
```

```
(4)[SibMe] BPM_ProcessServer_ME : [status = complete]
(5)[save and exit]
```

Please enter the number for the database component :5

```
[status] bpm.standard.nd is complete with 0 remaining item(s):
```

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with DB2 in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commond
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

```
Please enter the output directory of the SQL files to be generated for BPM_ProcessServer [default=] :
```

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 1 to select **(1)DB2-distributed**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
Schema name[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database provider for the **DB2-distributed** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 JCC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated.

After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
```

```
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
```

```
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
```

```
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
```

```
...
```

```
[WARNING] 2 potential problems are found in the scripts. They are
```

```
DB_USER @ line 46 in file configCommonDB.bat
```

```
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.



*Creating database design files and database scripts (Windows network deployment):*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files and database scripts for deployment environments with DB2 for z/OS:*

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory C:\tmp\DB2-zOS\W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as C:\tmp\DB2-zOS\W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as C:\tmp\DB2-zOS\W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root*\util\dbUtils directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```



[info] Please pick one of the following [design option(s)] :

- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(1) Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

[info] Please pick one of the following [database pattern(s)] :

- (1) bpm.advanced.nd.topology
- (2) bpm.advanced.standalone
- (3) bpm.standard.nd
- (4) bpm.standard.standalone
- (5) wesb.nd.topology
- (6) wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include “.nd.”

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1) bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2) [BPC] WBI\_BPC : [status = not complete]
- (3) [BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4) [BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI\_BSPACE : [status = not complete]
- (6) [SibME] WBI\_BPC\_ME : [status = not complete]
- (7) [SibME] WBI\_CEI\_ME : [status = not complete]
- (8) [SibME] WBI\_SCA\_APP\_ME : [status = not complete]
- (9) [SibME] WBI\_SCA\_SYS\_ME : [status = not complete]
- (10) [SibMe] BPM\_PerformanceDW\_ME : [status = not complete]
- (11) [SibMe] BPM\_ProcessServer\_ME : [status = not complete]
- (12) [save and exit]

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

[status] WBI\_CommonDB is not complete with 1 remaining item(s):  
[ 1 ] WBI\_CommonDB.WBI\_CommonDB : : DbType key is not set.

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type y and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

7. Type 2 to select **(2)DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Data source user name[default=] :
Database schema name (SQLID)[default=] :
Bufferpool of 4K size for tables[default=BP1] :
Bufferpool for indexes[default=BP2] :
Bufferpool of 4K size for LOB data[default=BP3] :
Bufferpool of 8K size for tables[default=BP8K1] :
Bufferpool of 16K size for tables[default=BP16K1] :
Bufferpool of 32K size for tables[default=BP32K1] :
Storage group name[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-zOS** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS
- (2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database Location on z/OS system[default=] :
Database server host[default=] :
Database server port[default=] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed
through jdbcDriver Type 4)[default=] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

**Note:** For DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. Typically, the Process Server, Performance Data Warehouse, and messaging engine components require unique schema names; the remaining components can be assigned the same

schema name. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).

- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.
- When configuring the Process Server component (**BPM\_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

- (1) false
- (2) true

If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

If the database is for a Process Server, you must choose the **false** option.

If you selected **bpm.advanced.nd.topology**, after you configure the master component, you must manually configure the **BPM\_PerformanceDW** and **BPM\_ProcessServer** components. Also edit the configuration values of the remaining components to ensure that the configuration is appropriate for each of the databases you want to create.

[info] Please pick one of the following [database component(s)] :

- (1) [WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2) [BPC] WBI\_BPC : [status = complete]
- (3) [BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4) [BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5) [BSPACE] WBI\_BSPACE : [status = complete]
- (6) [SibME] WBI\_BPC\_ME : [status = complete]
- (7) [SibME] WBI\_CEI\_ME : [status = complete]
- (8) [SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9) [SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10) [SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11) [SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12) [save and exit]

Please enter the number for the database component :12

[status] bpm.advanced.nd.topology is not complete with 6 remaining item(s):  
[ 1 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'schemaName' for SCHEMA is empty.  
[ 2 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'catalog' for VCAT is empty.  
[ 3 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'adminUserName' for SECURITY\_ADMIN\_USER is empty.  
[ 4 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects : required property 'adminPassword' for SECURITY\_ADMIN\_PASSWORD is empty.  
[ 5 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects : required property 'schemaName' for SCHEMA is empty.  
[ 6 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects : required property 'catalog' for VCAT is empty.

11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

generate database scripts? (y/n) [default=y] :

14. To generate the database scripts for each component in the directory that you specified earlier, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories. Each of the directories also contains a **createDB2.sh** script, which you can later run together with the database scripts to create the database objects.
- A log file named **dbDesignGenerator.log** is created in the directory from which you ran the **DbDesignGenerator** command.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment:*

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.

- The location of the JDBC driver on the system where the server profile will be created.
- The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :
```



- (1) Create a database design for Standalone profile or Deployment Environment
- (2) Create a database design for a single component
- (3) Edit an existing database design
- (4) Generate database scripts from a database design
- (5) exit [q]

Please enter the number for the design option :

2. To select the option **(2) Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1) bpc
- (2) bpcreporting
- (3) bpm\_performancedw
- (4) bpm\_processserver
- (5) bspace
- (6) cei
- (7) sca
- (8) sibme
- (9) wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4) bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You must specify a fully qualified directory path where you want to collate all the SQL scripts for a specific database; for example, WAS\_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB. Then press Enter. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1) DB2-distributed
- (2) DB2-zOS
- (3) Oracle
- (4) SQL Server

5. Type 2 to select **(2) DB2-zOS**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :  
 Data source user name[default=] :  
 Database schema name (SQLID)[default=] :  
 Bufferpool of 4K size for tables[default=BP1] :  
 Bufferpool for indexes[default=BP2] :  
 Bufferpool of 4K size for LOB data[default=BP3] :

Bufferpool of 8K size for tables[default=BP8K1] :  
Bufferpool of 16K size for tables[default=BP16K1] :  
Bufferpool of 32K size for tables[default=BP32K1] :  
Storage group name[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **DB2-z/OS** database type:

[info] Please pick one of the following [database provider(s)] :

(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS  
(2)DB2 Universal JDBC Driver Provider on local z/OS

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider (XA) on remote z/OS** as the database provider, type the number 1 and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.

Database Location on z/OS system[default=] :

Database server host[default=] :

Database server port[default=] :

Data source user name[default=] :

Data source password[default=] :

DB2 Universal JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/DB2] :

Universal JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/DB2] :

DB2 Universal JDBC driver native path(This should be empty since DB2 z on remote Machine accessed through jdbcDriver Type 4)[default=] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false**. This option enables you to run the SQL scripts manually at your designated time, and to create and organize the databases according to your DB2 for z/OS best practices and conventions.

After you complete the last prompt, you see information similar to the following example:

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:



generate database scripts? (y/n) [default=y] :

10. To generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
WAS_HOME/profiles/default/dbscripts/DB2-zOS/W8CELLDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

**Related tasks:**

“Creating the messaging engine data stores” on page 3373

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

*Generating scripts for creating database objects in DB2 for z/OS:*

From an installation of IBM Business Process Manager, you can generate database scripts for creating IBM Business Process Manager database objects in a DB2 for z/OS database subsystem. You can use the database design tool to generate the scripts.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Prepare the following information, which provides input to the database scripts that you plan to generate:

- Information about how IBM Business Process Manager and its components are installed.
- Information about the database configuration that you are designing for the DB2 for z/OS subsystem. This information can be supplied by the database administrator or solution architect, and can include details about the required parameters and properties; for example:
  - Database server details
  - Data source details
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

To generate the database scripts for creating your database objects, you can run the database design tool in silent mode with the **-g** command-line parameter. In this mode, the tool requires input from a database design file that defines the configuration for your DB2 for z/OS database. The database design file must already exist before you generate the database scripts. You can generate the database design file by running the database design tool in interactive mode, or by using a configuration planning spreadsheet:

- You can run the database design tool in interactive mode to generate a database design file. Follow the prompts in the tool to define a DB2 for z/OS database configuration for your stand-alone or network deployment topology.

- The configuration planning spreadsheet is a supplementary artifact that can be downloaded from the IBM Support Portal. The spreadsheet is primarily intended for use with IBM Business Process Manager for z/OS, but can be tailored for use with IBM Business Process Manager on Linux on System z. The configuration planning spreadsheet consists of a number of worksheets that use color coding and comments to assist with data entry, and to help you establish naming conventions. The spreadsheet also includes macros that you can use to generate a database design file, among other files. Before using the spreadsheet, run the database design tool in interactive mode to become familiar with the required configuration options for each of the components.

You can download the Microsoft Excel spreadsheet from Techdoc WP102075. The data that you require to complete the spreadsheet can span multiple roles, such as system programmer, product administrator, and database administrator. Therefore, it is advisable to review the spreadsheet in detail and then work with the relevant personnel to establish naming conventions and to complete the entries. After completing the entries, use the **Save** button on the DataBase Design File worksheet to generate the database design file.

While defining a DB2 for z/OS database configuration, you can specify how to group the SQL database scripts that you want to generate for the IBM Business Process Manager components. The default setting is for the scripts to be generated in individual directories for each component. Before you run the database design tool, decide on a location for storing the files that the tool will generate. It is important to use a directory path and structure that reflects how you want the database objects to be distributed across the DB2 for z/OS subsystem. For each database to be created, use a single output directory, and generate all the schema that is required to create the database, into that directory. As a reference aide, you can also base the name of the SQL directory on the name of the database that you plan to create.

For example, if you intend to create a single database called W1MYDB, you can generate the database scripts for all the database components in the single directory C:\tmp\DB2-zOS\W1MYDB. If you require two databases, which are scoped at the cell level and the cluster level, you can structure the database schema as shown in the following example:

- To create a database called W8CELLDB, which contains database objects that are scoped at the cell level, generate the database scripts for the Common database component in a directory such as C:\tmp\DB2-zOS\W8CELLDB. You can later run the generated schema to create the database objects for the Common database component in the W8CELLDB database.
- To create a database called W8S1DB, which contains the database objects that are scoped at the cluster level, generate the database scripts for all the remaining IBM Business Process Manager components in a directory such as C:\tmp\DB2-zOS\W8S1DB. You can later run the generated schema to create the database objects for these components in the W8S1DB database.

1. If necessary, copy the database design file, which defines your DB2 for z/OS configuration, to the workstation where IBM Business Process Manager is installed, to make the file available for use with the **DbDesignGenerator** command.
2. From the command line, change to the IBM Business Process Manager directory where the **DbDesignGenerator** command is stored:

```
cd install_root\util\dbUtils
```

For example: cd C:\IBM\WebSphere\AppServer\util\dbUtils

3. Run the **DbDesignGenerator** command using the following syntax:

```
DbDesignGenerator.bat -g db_design_file
```

where *db\_design\_file* is the fully qualified name of the database design file. For example:

```
DbDesignGenerator.bat -g C:\tmp\DB2-zOS\W8CELL.nd.dbDesign
```

4. Review the information that is displayed on the screen, checking that no error messages are displayed. For each database component, the database design tool writes information messages to the screen to indicate that the scripts were generated in the directory specified in the database design file. For example, for the Common database component, you see information similar to the following example:

[info] The script(s) have been generated in  
C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB for WBI\_CommonDB

After the last set of scripts is generated, you see information similar to the following example:

[info] thanks, quitting now ...

A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

**Tip:** In the database design file, the directories for the SQL scripts are defined by the `ddl_outDir` variable in each component section. If you want to generate the SQL scripts to a set of different directories, a quick way to do this is to manually update the values of the `ddl_outDir` variables. Then save the database design file, and run the **DbDesignGenerator** command again.

1. Check the specified directory locations to see the database scripts that were generated. Each directory additionally includes a **createDB2.sh** script that you can use to run the SQL.

To use these scripts to create the IBM Business Process Manager database objects in the DB2 for z/OS subsystem, you must first transfer the scripts to the z/OS system.

2. Use FTP to transfer all the generated database scripts to the z/OS system that contains the installation of DB2. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
3. Run the **createDB2.sh** script from the z/OS UNIX System Services command environment to create the DB2 for z/OS database objects.

**Note:** The **createDB2.sh** script requires execute permissions.

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-zOS-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

### *Setting up databases for Oracle:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

You can use a single instance of Oracle for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use different user IDs for the three different BPM databases.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```

2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

### *Creating database design files for Oracle:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Creating database design files for deployment environments with Oracle:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.

- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.
4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)bpm.standard.nd
(4)bpm.standard.standalone
(5>wesb.nd.topology
(6>wesb.standalone
```

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include ".nd."

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BPC] WBI_BPC : [status = not complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = not complete]
(6)[SibME] WBI_BPC_ME : [status = not complete]
(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:



```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type **y** and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

7. Type **3** to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name(SID) [default=CMNDB] :
Database User name[default=] :
Database schema [default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
Database Location(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter **'s'**; or enter anything else to continue :

9. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

[info] Please pick one of the following [database provider(s)] :

(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=1521] :
```

[info] Please pick one of the following [Oracle driver type(s)] :

```
(1)oci8
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
Data source user name[default=] :
Data source password[default=] :
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

-----

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components,
since other components may inherit values from them.
```

[info] Please pick one of the following [database component(s)] :

```
(1)[WBI_CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPC] WBI_BPC : [status = complete]
(3)[BPM_PerformanceDW] BPM_PerformanceDW : [status = complete]
(4)[BPM_ProcessServer] BPM_ProcessServer : [status = not complete]
(5)[BSPACE] WBI_BSPACE : [status = complete]
(6)[SibME] WBI_BPC_ME : [status = complete]
(7)[SibME] WBI_CEI_ME : [status = complete]
(8)[SibME] WBI_SCA_APP_ME : [status = complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

10. Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.



If you selected **bpm.advanced.nd.topology**, you must manually configure the **BPM\_PerformanceDW**, **BPM\_ProcessServer**, and the corresponding **SibMe** components for the required database authentication.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :3

-----  
[status] BPM\_PerformanceDW is not complete with 1 remaining item(s):  
[ 1 ] BPM\_PerformanceDW.BPM\_PerformanceDW : databaseObjects :  
required property 'databaseUser' for DB\_USER is empty.

Edit this database component? (y/n) [default=y] :

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]
- (9)[SibME] WBI\_SCA\_SYS\_ME : [status = complete]
- (10)[SibMe] BPM\_PerformanceDW\_ME : [status = complete]
- (11)[SibMe] BPM\_ProcessServer\_ME : [status = complete]
- (12)[save and exit]

Please enter the number for the database component :4

-----  
[status] BPM\_ProcessServer is not complete with 3 remaining item(s):  
[ 1 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'databaseUser' for DB\_USER is empty.  
[ 2 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminUserName' for SECURITY\_ADMIN\_USER is empty.  
[ 3 ] BPM\_ProcessServer.BPM\_ProcessServer : databaseObjects :  
required property 'adminPassword' for SECURITY\_ADMIN\_PASSWORD is empty.

Edit this database component? (y/n) [default=y] :

- 11. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :

- 12. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :

13. Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

14. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\Oracle-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.
- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with DB2 in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.

- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
```

- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

[info] Please pick one of the following [component(s)] :

- (1)bpc
- (2)bpcreporting
- (3)bpm\_performancedw
- (4)bpm\_processserver
- (5)bspace
- (6)cei
- (7)sca
- (8)sibme
- (9>wbi\_commondb

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 3 to select **(3)Oracle**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name(SID) [default=CMNDB] :

Database User name[default=] :

Database schema [default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

Database Location(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **Oracle** database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)Oracle JDBC Driver # XA data source # Oracle JDBC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
```

```
Database server host[default=] :
```

```
Database server port[default=1521] :
```

```
[info] Please pick one of the following [Oracle driver type(s)] :
```

```
(1)oci8
```

```
(2)thin
```

```
Please enter the number for the Oracle driver type(s)? [default=thin] :
```

```
Data source user name[default=] :
```

```
Data source password[default=] :
```

```
Oracle JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
```

```
] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\Oracle-CommonDB for WBI_CommonDB
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

### Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

### Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v Oracle-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

### Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

*Creating databases and database design files for SQL Server:*

IBM Business Process Manager requires a Process Server database, Performance Data Warehouse database, and Common database. The Common database contains Business Space and other components. You must create the databases before network deployment.

**Tip:** If you are using the federated repositories as a user registry, you can ignore warnings in the `systemout.log` file about maximum key length: ... **Warning! The maximum key length is 900 bytes ....** If you are using the stand-alone LDAP registry, ensure that the number of characters in all the user distinguished name (DN) entries in your organization do not exceed the 131 character limit. If the number of characters in any of the user DN entries exceeds 131 characters, you must change the user account registry to the federated repositories option.



## Creating SQL Server databases:

You can create the required databases before you create profiles.

The default database names are BPMDB for the Process Server database, PDWDB for the Performance Data Warehouse database, and CMNDB for the Common database.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be configured on the same database as the other BPM components.

1. Install Microsoft SQL Server.
2. Use the following commands to create the databases for SQL Server:
  - a. To create the Process Server (BPMDB) and Performance Data Warehouse (PDWDB) databases:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db\_user\_account** and **db\_user\_password** are the user name and password that are used to log in to create the database, and **database\_name** is the name of the database that you are creating. **COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS** is included because these databases must be case-insensitive.

- b. To create the Common database:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CS_AS"
```

The difference is the COLLATE clause. CMNDB requires case-sensitive collation.

**Important:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardi software.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```

com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
 at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
 at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
 at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
 at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
 at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)

```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is *dbo* and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

- a. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
- b. From the Design view, press F4 to view the Properties window.
- c. From the Properties window, update the schema name.
- d. Right-click the tab and select **Close** to close the Design view.
- e. Click **OK** when prompted to save. The selected table is transferred to the schema.
- f. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 202.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to `READ_COMMITTED`. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

#### Configuring XA transactions:

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the `SQLServerXADataSource` class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'."**



1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

**On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

**On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.
- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
- f. Click **OK** again to close the Properties window, and then close Component Services.
- g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

2. Configure the JDBC Distributed Transaction Components:

- a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
- b. Extract the archive to any folder.
- c. Copy the sqljdbc\_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc\_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc\_xa.dll file in the x64 folder.
- d. Run the xa\_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc\_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
- e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

*Creating database design files for SQL Server:*

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating database design files for deployment environments with SQL Server:*

You can use the database design tool to generate a design file that can be used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool optionally generates database scripts that you can use to create your database tables.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
 edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
 when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
 When a db_scripts_output_directory is given, the database scripts
 in the specified directory will be validated. Currently only
 scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
 [db_design_fileN] [-d output_directoryN]
 generate the database scripts from the specified design files in batch mode.
 The generated scripts will be put in the corresponding output
 directories or the default locations if output directories are absent.
```

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Restriction:** If you want to use a different Business Space database when you run the database design tool, you cannot use BSPACE as the database name. If you need the database name to be BSPACE, you can complete the following the steps:

1. Run the database design tool as described in the procedure and accept the default database name selected for Business Space.

**Note:** Complete the steps to create a database design file, but do not generate the database scripts when prompted.

2. Edit the database design file that is generated, and update the database name for Business Space in the **[begin] = BSpace : WBI\_BSPACE** section, changing the **databaseName** value to **BSPACE**.
3. Save the database design file.

4. Run the database design tool again with the option to generate database scripts from a database design, and use the updated database design file to provide the input for the database scripts.

If you use a database design file that designated BSPACE for the Business Space database, when you create a deployment environment or a stand-alone profile using that database design file, you must manually change the Business Space database name back to BSPACE in the Deployment Environment wizard or in the Profile Management Tool.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

- (1)Create a database design for Standalone profile or Deployment Environment
- (2)Create a database design for a single component
- (3)Edit an existing database design
- (4)Generate database scripts from a database design
- (5)exit [q]

Please enter the number for the design option :

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

You are prompted to choose a database pattern, as shown in the following example:

```
[info] Please pick one of the following [database pattern(s)] :
```

- (1)bpm.advanced.nd.topology
- (2)bpm.advanced.standalone
- (3)bpm.standard.nd
- (4)bpm.standard.standalone
- (5>wesb.nd.topology
- (6>wesb.standalone

3. To create a database design pattern for the deployment environment that you plan to configure, type the number for the appropriate option and press Enter. Select options that include ".nd."

For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see the list of database components that you can configure for the selected environment, with a prompt to choose a component to configure; for example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]
- (2)[BPC] WBI\_BPC : [status = not complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = not complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = not complete]
- (6)[SibME] WBI\_BPC\_ME : [status = not complete]

```

(7)[SibME] WBI_CEI_ME : [status = not complete]
(8)[SibME] WBI_SCA_APP_ME : [status = not complete]
(9)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(10)[SibMe] BPM_PerformanceDW_ME : [status = not complete]
(11)[SibMe] BPM_ProcessServer_ME : [status = not complete]
(12)[save and exit]

```

4. Type the number for the appropriate option to configure the master database component, and press Enter. The database component listed as the master component displays **[master]** beside the name and must be configured first.

For example, type the number 1 to select option **(1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = not complete]**, and press Enter. You see information similar to the following example:

```

[status] WBI_CommonDB is not complete with 1 remaining item(s):
[1] WBI_CommonDB.WBI_CommonDB : : DbType key is not set.

```

Edit this database component? (y/n) [default=y] :

5. To edit the database configuration for the component, type *y* and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for WBI\_CommonDB [default=] :

6. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

```

(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server

```

7. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

```

[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database server host[default=] :
Database User name[default=] :
Database schema[default=] :
System user name(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile
creation.)[default=] :

```

8. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

9. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

(1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the data source properties section.  
Database server port[default=1433] :

[info] Please pick one of the following {Do you want to use windows authentication for the databases ? (s)} :

- (1) false
- (2) true

Data source user name[default=] :  
Data source password[default=] :  
MS SqlServer JDBC driver path[default=\${WAS\_INSTALL\_ROOT}/jdbcdrivers/SQLServer] :

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

[status] WBI\_CommonDB is complete with 0 remaining item(s):

-----  
[info] Please edit any database component with status of 'not complete' for required properties.  
[info] Completed database components can be edited to change existing or defaulted property values.  
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

- (1)[WBI\_CommonDB] WBI\_CommonDB : [master] [status = complete]
- (2)[BPC] WBI\_BPC : [status = complete]
- (3)[BPM\_PerformanceDW] BPM\_PerformanceDW : [status = complete]
- (4)[BPM\_ProcessServer] BPM\_ProcessServer : [status = not complete]
- (5)[BSPACE] WBI\_BSPACE : [status = complete]
- (6)[SibME] WBI\_BPC\_ME : [status = complete]
- (7)[SibME] WBI\_CEI\_ME : [status = complete]
- (8)[SibME] WBI\_SCA\_APP\_ME : [status = complete]

```
(9)[SibME] WBI_SCA_SYS_ME : [status = complete]
(10)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(11)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(12)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

- Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

If you selected **bpm.standard.nd**, after you configure the **BPM\_ProcessServer**, you must manually configure the **BPM\_PerformanceDW** component for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW : [status = not complete]
(3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(5)[save and exit]
```

```
Please enter the number for the database component :2
```

```

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[1] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects :
required property 'databaseUser' for DB_USER is empty.
```

```
Edit this database component? (y/n) [default=y] :
```

- When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, type the appropriate number to select **[save and exit]**, and press Enter. You see information similar to the following example:

```
[status] bpm.advanced.nd.topology is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

- Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

- Press Enter to accept the default name for the database design file, or type another file name and then press Enter. If a file of the same name already exists in the specified directory, you can choose to either overwrite the file or specify a different file name. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

- Optional: If you also want to generate database scripts based on the information provided to the database design tool, type **y** and press Enter. A series of information messages is displayed to indicate that the scripts for all the components have been generated. For each component, you see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\SQLServer-CommonDB for WBI_CommonDB
```

After the last set of scripts is generated, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

The following files are created:

- The database design file is created in the specified directory.



- The database scripts are created in one or more specified directories.
- A log file named `dbDesignGenerator.log` is created in the directory from which you ran the **DbDesignGenerator** command.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- When you use the Profile Management Tool to create a profile
- When you use the **manageprofiles** command-line utility to create a profile
- When you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

*Creating database design files for specific components with SQL Server in a network deployment environment:*

You can use the database design tool to generate a design file and database scripts for the database tables required by specific IBM Business Process Manager components.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator or solution architect. Alternatively, it might be a description of required parameters and properties. This information must include:
  - The location of the database tables.
  - The location of the JDBC driver on the system where the server profile will be created.
  - The user ID and password for authenticating to the database.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles that you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

You can run the database design tool in interactive mode to define your database configuration values and save them to a new database design file or to an existing file. You can optionally generate database scripts that can be used to create the database objects for your database type. When you run the tool, the available options change depending on the environment and database type being configured.

You also can run the database design tool with command-line parameters to edit or validate an existing database design file, or to generate database scripts for the configuration defined in an existing database design file. The **DbDesignGenerator** command has the following command-line parameters:

```
-? , -help
 display help info.

-e db_design_file_name
```



edit the specified database design file (e.g. \*.dbDesign, \*.properties).

-v db\_design\_file | db\_scripts\_output\_directory  
when a db\_design\_file is given, validation will be done on the specified database design file based on the database specs.  
When a db\_scripts\_output\_directory is given, the database scripts in the specified directory will be validated. Currently only scripts generated from template ddl generator can be validated.

-g db\_design\_file [-d output\_directory] [db\_design\_file2] [-d output\_directory2] ... [db\_design\_fileN] [-d output\_directoryN]  
generate the database scripts from the specified design files in batch mode.  
The generated scripts will be put in the corresponding output directories or the default locations if output directories are absent.

**Restriction:** The database design tool does not support Common Event Infrastructure (CEI).

**Important:** If you intend to generate database scripts for the service integration bus messaging engines, you must repeat all the steps of the following procedure to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times.

1. Run the **DbDesignGenerator.sh** command from the *install\_root\util\dbUtils* directory. For example:

```
C:\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat
```

**Tip:** If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

Please enter the number for the design option :

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

You are prompted to choose a component, as shown in the following example:

```
[info] Please pick one of the following [component(s)] :

(1)bpc
(2)bpcreporting
(3)bpm_performancedw
(4)bpm_processserver
(5)bspace
(6)cei
(7)sca
(8)sibme
(9>wbi_commond
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm\_processserver**, and press Enter.

You are prompted for an output directory for the SQL scripts that the database design tool can generate, and which you can use to create your database objects. If you choose to generate SQL scripts later in this procedure, the scripts are automatically saved to this directory.

Please enter the output directory of the SQL files to be generated for BPM\_ProcessServer [default=] :

4. Type an output directory for the SQL scripts and press Enter. You can specify a fully qualified directory path, or a path that is relative to the current directory from which the database design tool is being run. If you have not yet created the directory in your file system, the database design tool automatically creates it when the SQL scripts are generated. If the specified directory already exists, you can choose to either overwrite its contents or specify another directory.

**Tip:** The value that you specify will be saved as a reference in the database design file when this file is generated.

You are prompted to select the database type that you are configuring; for example:

[info] Please pick one of the following [database type(s)] :

- (1)DB2-distributed
- (2)DB2-zOS
- (3)Oracle
- (4)SQL Server

5. Type 4 to select **(4)SQL Server**, and press Enter. You obtain a series of prompts for the database properties. These prompts vary based on the database type.

For example, after you select this database type for configuration of the Common database, you see a series of prompts similar to the following example:

[info] Please enter the values for the properties in the database objects section.

Database name[default=CMNDB] :

Database server host[default=] :

Database User name[default=] :

Database schema[default=] :

System user name(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

6. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter.

The administrative security prompt (if present) refers to the user name and password for the administrative console.

After you complete the last prompt for the database properties, you see information similar to the following example:

[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :

7. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

**Tip:** If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the **SQL Server** database type:

[info] Please pick one of the following [database provider(s)] :

- (1)Microsoft SQL Server JDBC Driver # XA data source # Microsoft SQL Server JDBC Driver (XA)

- a. Type the appropriate number to select a database provider for the data source, and press Enter. After you select this database provider, you see a series of prompts similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server port[default=1433] :
```

```
[info] Please pick one of the following {Do you want to use windows authentication for the
databases ? (s)} :
```

- (1) false
- (2) true

```
Data source user name[default=] :
Data source password[default=] :
MS SqlServer JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer] :
```

**Note:** The password is encrypted in the generated output files.

- b. At each prompt, either press Enter to accept the default value shown, or type your preferred value and press Enter. If you are asked whether you want to create the database tables during configuration, specify **false** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

8. Press Enter to accept the default directory for the database design file, or type another directory to which you want to save the file and then press Enter. If you have not yet created the directory in your file system, the database design tool will automatically create it when the file is generated. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
```

9. Press Enter to accept the default file name, or type another file name and then press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

10. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter. You see information similar to the following example:

```
[info] The script(s) have been generated in
C:\IBM\WebSphere\AppServer\util\dbUtils\SQLServer-CommonDB for WBI_CommonDB
```

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the locations that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

*Troubleshooting the database design tool:*

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

## Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[1] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[2] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

## Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v SQL Server-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

## Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

## Configuring the network deployment environment using the `configureNode` command:

To configure a typical network deployment environment, you can use the **configureNode** command instead of using the Profile Management Tool and Deployment Environment wizard or the **manageprofiles** and **wsadmin** commands. This command provides limited customization by using a properties file, and creates a ready-to-use environment. You can also use **configureNode** later to extend the deployment environment.

**Restriction:** This command is not supported if your database type is DB2 for z/OS.

**Important:** The **configureNode** command cannot be run twice on one machine without first deleting the profile that was created. To run the **configureNode** command a second time, first delete the profile.

*Creating the deployment environment using the `configureNode` command:*

You can use the **configureNode** command to create a typical network deployment environment. The goal of this command is to provide a ready-to-use environment. The only parameters that can be customized are provided in a properties file.

**Restriction:** This command is not supported if your database type is DB2 for z/OS.

You must have installed the product and created all the databases that are specified in the properties file.

**Important:** You must have already installed the database software and created the databases (but do not create tables). You must also have created all the users that you specify in the properties file. Make sure the users have the installation privileges listed in Database privileges.

For your SQL Server database server:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.

- If connections to the database will be made by the current Windows user that the server is running under, the SQL Server must have **Windows authentication mode** or **SQL Server and Windows Authentication mode** enabled, as specified through Microsoft SQL Server Management Studio.

**Restriction:** Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.

The **configureNode** command uses default ports to create the deployment manager. You cannot customize the ports using this command. The parameters listed in the sample files are the only parameters that can be customized. Make sure that no other profiles that are configured with default ports are started on the same machine.

The **configureNode** command creates the following network deployment topology:

- A single cell environment
- The environment contains the number of cluster members that you specify in the properties file
- The environment uses the Remote Messaging, Remote Support, and Web topology pattern, which includes the following four clusters:
  - Messaging infrastructure cluster
  - Supporting infrastructure cluster
  - Application deployment target cluster
  - Web application cluster

The command performs the following tasks:

- Creates the deployment manager node based on the values in the deployment manager properties file and starts the deployment manager.
- Creates the deployment environment definition.
- Creates a custom node based on the values in the custom node properties file.
- Federates the node and adds the node to the deployment environment.
- Generates the deployment environment.
- Creates the database tables. (You must already have created the databases. See "Before you begin" for instructions.)
- Runs the bootstrap utility to load the Process Server database with system information.
- Configures Business Space and Process Portal on the web application cluster.

The default port is used to create the deployment manager if the optional ports are not specified in the properties file. If a deployment manager is already running on the machine when you run **configureNode**, creating the deployment environment fails. If a deployment manager has already been created but is not running, you must change the port of one of the deployment managers or you might get a port conflict error when you start the other deployment manager.

To create the deployment environment for the first time, complete the following steps:

1. On the machine where you want to create the deployment environment, locate the sample properties files: *install\_root\util\ndUtils\samples\*.
2. Copy the sample files and modify the files to reflect your environment. For example, for a deployment manager for IBM BPM Advanced for Process Server, choose the *sample\_adv\_pc\_dmgr.properties* file.

For each database that you want to create, copy the database parameters section and specify the database name, user name, password, and schema name. The Process Server database, Performance Data Warehouse database, and Common database are required, and they must have different names. For IBM BPM Advanced, the Business Process Choreographer database is also required.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

**Restriction:** The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the names of the two files you created. The custom node file is optional, but if you do not include it, only the deployment manager is created. For example:

```
install_root\util\ndUtils\configureNode.bat -dmgr_response my_dmgr_response_file.properties
-response node01_response_file.properties
```

**Note:** If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to **127.0.0.1 localhost** in the hosts file. For example: **127.0.0.1 localhost example.ibm.com**.

You now have a deployment manager, a custom node, and a deployment environment that use the Remote Messaging, Remote Support, and Web topology pattern.

Messages are recorded in the file *install\_root/logs/config/configureNode.log*.

**Tip:** After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception `com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT`. This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

**Important:** If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

**Related concepts:**

 [Virtual hosts](#)

**Related tasks:**

 [Configuring virtual hosts](#)

**Related reference:**

 [Test connection service](#)

*Extending the deployment environment using the configureNode command:*

After you have created a deployment environment using the **configureNode** command, you can run the command at a later time to extend the deployment environment.

You need the following prerequisites before running the **configureNode** command:

- You must have installed the product on the machine where you want to extend the deployment environment.
- The deployment manager must be running.
- You must have created the deployment environment by running the **configureNode** command.

**Restriction:** Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.



1. On the machine where you want to extend the deployment environment, locate the sample custom node files: `install_root\util\ndUtils\samples\`
2. Copy the sample file that is most similar to the properties file that you need. For example, to extend the deployment environment for IBM BPM Advanced for Process Server, choose the `sample_adv_ps_node.properties` file.

Modify the new custom node file to reflect your environment, including the details of the deployment manager.

For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

**Restriction:** The parameters listed in the sample files are the only parameters that can be customized.

3. Run the **configureNode** command, passing it the name of the sample file. For example:

```
install_root\util\ndUtils\configureNode.bat -response node01_response_file.properties
```

**Note:** If you receive the **Failed to perform Security setting update** error, check to ensure that the global IP address has not been appended to **127.0.0.1 localhost** in the hosts file. For example: **127.0.0.1 localhost example.ibm.com**.

You have extended the deployment environment.

Messages are recorded in the file `install_root/logs/config/configureNode.log`.

**Tip:** After you have configured a network deployment environment, if you test the data store connection (for example, in the administrative console, on the page **Resources > JDBC > Data sources**), you get a message saying that the test connection operation failed with the exception `com.ibm.wsspi.runtime.variable.UndefinedVariableException: Undefined Variable WAS_INSTALL_ROOT`. This does not necessarily indicate that there will be a problem accessing the data source at run time. Ensure that the location of your JDBC driver files is accessible to every client that must use the data source, and configure the variable with the full path of that location. Disregard the test connection error unless you are also experiencing trouble connecting to the data store at run time. For additional information, see the WebSphere Application Server documentation about the test connection service.

**Important:** If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

#### Related concepts:

 [Virtual hosts](#)

#### Related tasks:

 [Configuring virtual hosts](#)

#### Related reference:

 [Test connection service](#)

### Configuring profiles and creating a network deployment environment:

After you install the product, you must create or augment a deployment manager and one or more custom profiles to define the runtime environment. Before starting the deployment manager, you must have configured the databases that are to be used with IBM Business Process Manager.



*Creating or augmenting network deployment profiles:*

If you used the **configureNode** command to create and configure your deployment environment, no further configuration is required. Otherwise, you must create or augment a deployment manager profile and one or more custom profiles before creating the deployment environment. Using profiles, you can have more than one runtime environment on a system, without having to install multiple copies of IBM Business Process Manager.

*Creating or augmenting network deployment profiles with a DB2 database server:*

You can configure a network deployment environment for IBM Business Process Manager using a DB2 database server.

*Preparing the DB2 database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components. You can either create a new database and tables or postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

If you plan to configure the Common database on a remote server, you must have created the database before beginning to create or augment the profile. You can create a database on the local server or use an existing one on a remote server.

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:**  

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSPProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.

- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

### IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 203. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.



Table 203. Required Common database configuration fields for DB2 Database (continued)

Field	Action needed
Server port	Accept the default value of 50000 or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with a DB2 database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:****Vista****Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.



- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.
11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.  
If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The

password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the

advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

**IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - e. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 204. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdriivers/DB2</code> .

Table 204. Required Common database configuration fields for DB2 Database (continued)

Field	Action needed
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

20. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
21. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create a deployment manager profile.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.

- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:
    - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
    - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
    - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
    - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
    - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
  2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response

- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.



Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
```



```

When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.

```

```

Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign

```

```

#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbidbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:

```

```

The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#
Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcd drivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has

```



```

privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their

```

```

production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#

```

```

Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr
#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#

```

```

Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod

```

```

#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile

```

```

#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:

```

```

localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true

```



```

#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:

```

```

Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService

```

```

#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None

```



```

#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
true
Default Values:
None
#####
dbCreateNew=true

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbHostName
#
Description:
The database server host name or IP address. The default value is the
local host name. Required when the -bpmdbDesign parameter was not
specified. (It is not required if the -bpmdbDesign parameter is
specified.)
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbServerPort
#

```

```

Description:
The database server port number. Depending on the database you are using,
you can specify a different port number instead of the default port
number.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#

```

```

Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
isDeveloperServer=false

```

Sample response file for a Standard Process Server deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
```

```

Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.

```

```

The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#

```

```

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies

```



```

Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service

```

```

The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a

```

```

stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with a DB2 database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. **Advanced:** On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  10. **Advanced:** On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.

11. **Required:** On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all

internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:



- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.



- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 205. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of <code>localhost</code> or enter the correct database server host name.
Server port	Accept the default value of <code>50000</code> or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always `BPEDB`. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with a DB2 database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
- Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSPProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager and create a Business Process Choreographer sample configuration.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system. When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for

commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.
- Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the



service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **DB2**.
  - b. Select **Create a new local database** or **Use an existing local or remote database**. If you select to create a new database, the Profile Management Tool will create a new DB2 database.
  - c. Optional: If you want to change the default value, enter a new name for the Common database. The database name must be unique.
  - d. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - e. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 206. Required Common database configuration fields for DB2 Database

Field	Action needed
User name to authenticate with the database	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in <code>\${WAS_INSTALL_ROOT}/jdbcdrivers/DB2</code> .
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 50000 or enter the correct server port number.
Schema name	Enter a schema name for the Common database.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. Advanced: On the Business Process Choreographer Configuration page, select whether to create a sample Business Process Choreographer configuration. The sample configuration is for development use only.

**Important:** The sample Business Process Choreographer configuration uses its own schema within the Common database (CMNDB) and the schema name is always BPEDB. If these conditions are too restrictive, disable the Business Process Choreographer configuration during profile creation and configure Business Process Choreographer after you create the profile.

22. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
23. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the `manageprofiles` command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the `runas` command. Remember to put double quotation marks around the `manageprofiles` command and all parameters. For example, the following command can be run from the `install_root\bin` directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

If you installed BPM in a Windows domain controller environment and installed DB2 Express as part of installation, profile creation will fail because you cannot use the domain user ID to create a new database and tables. Before creating a profile, use one of the following solutions:

- Log into the Windows system as a local system user to create a profile.
  - Follow the steps listed in DB2 log file error: SQL1092N "USERID does not have the authority to perform the requested command or operation" before creating a profile.
1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the `install_root\properties\profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
  2. Find the appropriate template to use for the augmentation.



Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir*/profileTemplates/BPM directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response

- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

#### *Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

## Creating custom profiles using the Profile Management Tool:

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.

7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.
9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones.

If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the **manageprofiles** command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
- managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
- managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.



- managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response



- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
```

```

instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces

```

```

or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNameewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

```

```

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#

```

```

Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER

```

```

#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:

```



```

None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when

```

```

creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation

```

```

root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr
#####

Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01
#####

Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies

```

```

Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:

```

```

An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The

```

```

localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic

```

```

disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#

```



```

Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.

```

```

#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses DB2.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.

```

```

Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsrvr.adv
#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName

```

```

#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#

```

```

Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####

```

```

#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2_DATASERVER
Default Values:
DB2_DATASERVER
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The

```



```

host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing

```

```

environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses DB2.
#
Depending on your environment, you may need to change the default values.

```

```

#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:

```

```

true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to

```

```

set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck

```

```

#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.

```



```

This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.

```

```

Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:**

**Vista**

**Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### **Related information:**

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the `install_root\bin` directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.



1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install\_root\properties\profileRegistry.xml* file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
 Templates for each profile are located in the *install\_root\profileTemplates\BPM* directory for BPM templates and under *install\_root\profileTemplates* for other products. The following templates are available for custom profiles:
  - *managed.procctr*: for a Process Center custom profile for IBM BPM Standard.
  - *managed.procctr.adv*: for a Process Center custom profile for IBM BPM Advanced.
  - *managed.procsvr*: for a Process Server custom profile for IBM BPM Standard.
  - *managed.procsvr.adv*: for a Process Server custom profile for IBM BPM Advanced.
  - *managed.esbserver*: for a WebSphere Enterprise Service Bus custom profile.
3. Use the *augment* parameter to make changes to an existing profile with an augmentation template. The *augment* parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root\BPM\samples\manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- *PC\_Std\_DMGr\_DB2.response*
- *PC\_Std\_DMGr\_DB2z0S.response*
- *PC\_Std\_DMGr\_Oracle.response*
- *PC\_Std\_DMGr\_SQLServer.response*
- *PC\_Adv\_DMGr\_DB2.response*
- *PC\_Adv\_DMGr\_DB2z0S.response*
- *PC\_Adv\_DMGr\_Oracle.response*
- *PC\_Adv\_DMGr\_SQLServer.response*
- *PC\_Std\_Managed\_DB2.response*
- *PC\_Std\_Managed\_DB2z0S.response*
- *PC\_Std\_Managed\_Oracle.response*
- *PC\_Std\_Managed\_SQLServer.response*
- *PC\_Adv\_Managed\_DB2.response*
- *PC\_Adv\_Managed\_DB2z0S.response*
- *PC\_Adv\_Managed\_Oracle.response*
- *PC\_Adv\_Managed\_SQLServer.response*
- *PC\_Std\_StandAlone\_DB2.response*
- *PC\_Std\_StandAlone\_DB2z0S.response*



- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
  - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:  

```
profile_root\bin\startManager.bat
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile:

```
profile_root\bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMND1D2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### **Related information:**

 Add managed node settings (WebSphere Application Server)

*Creating or augmenting network deployment profiles with a DB2 for z/OS database server:*

You can configure a network deployment environment for IBM Business Process Manager using a DB2 Universal Database for z/OS database server.

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Creating Process Center deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool:

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.
- Click **Next**.
9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page
- When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.
11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.
- If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing

certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with a DB2 for z/OS database server using the Profile Management Tool on Linux:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```



For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:**

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
5. On the Profile Creation Options page, choose **Advanced** and click **Next**.
6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
7. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one



profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the

service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
15. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
16. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
17. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
18. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.

- dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the *install\_root\BPM\samples\manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response

- PS\_Adv\_DMgr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db2user -dbPassword db2secret
-dbType DB2_DATASERVER -procSvrDbName BPMDB -perfDWDName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
```



```

Parameter: profilename
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber

```



```

where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

```

```

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#

```

```

Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

```

```
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
```

```
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false
```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
```



```

(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMgr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:

```

```

profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies

```

```

#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option

```

```

set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:

```

```

Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.

```

```

#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on

```



```

the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

```

```

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

```

```

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.

```

```

Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux

```

```

service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#

```

```

Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:

```

```

A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=true

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.

```



```

#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#####
Parameter: dbStorageGroup
#
Description:
The storage group name for DB2 z/OS databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbConnectionLocation
#
Description:
The location of DB2 for z/OS database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####

```

```
environmentType=Test
```

```

Parameter: isDeveloperServer

Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.

Valid Values:
false
true
Default Values:
false

#isDeveloperServer=false
```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.

END COPYRIGHT

These response file has the applicable parameters for creating a
DMgr PS Std profile that uses DB2zOS.

Depending on your environment, you may need to change the default values.

To create a profile with this response file specify:

was.install.root/bin/manageprofiles -response PS_Std_DMGr_DB2zOS.response

If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .

Parameter: create
#
```

```

Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:

```

```

if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root

```

```

signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType

```

```

#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.

```



```

#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting deployment manager profiles for Process Center with a DB2 for z/OS database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
8. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:

- In the **Path** field, enter the directory path to the existing certificate.
- In the **Password** field, enter the password for the certificate
- In the **Keystore type** field, select the keystore type for the certificate you are importing.
- In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
- Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

12. **Advanced:** On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root\properties\portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

14. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

15. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.



- Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with a DB2 for z/OS database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

To use a DB2 for z/OS database, you must choose **Advanced** profile creation and specify a database design file.

1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.

5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose **Advanced** and click **Next**.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
- The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

7. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
8. Advanced: On the Profile Name and Location page, perform the following steps:
  - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
  - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
  - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
9. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.Click **Next**.
10. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.
11. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.



- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

12. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

13. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root\properties\portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

14. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

15. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.

With a DB2 for z/OS database, you cannot run the database scripts automatically as part of the profile creation process.
  - d. Click **Next**. The design file location is passed to the command line to complete the database configuration.
16. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
17. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
18. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
19. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Run the scripts to create and configure the DB2 for z/OS database.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the `manageprofiles` command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the `install_root\bin` directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the `install_root\properties\profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the `install_root\profileTemplates\BPM` directory for BPM templates and under `install_root\profileTemplates` for other products. The following templates are available:
  - `dmgr.procctr`: for a Process Center deployment manager profile for IBM BPM Standard.
  - `dmgr.procctr.adv`: for a Process Center deployment manager profile for IBM BPM Advanced.
  - `dmgr.procsvr`: for a Process Server deployment manager profile for IBM BPM Standard.
  - `dmgr.procsvr.adv`: for a Process Server deployment manager profile for IBM BPM Advanced.
  - `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_Standalone\_DB2.response
- PC\_Std\_Standalone\_DB2zOS.response
- PC\_Std\_Standalone\_Oracle.response
- PC\_Std\_Standalone\_SQLServer.response
- PC\_Adv\_Standalone\_DB2.response
- PC\_Adv\_Standalone\_DB2zOS.response
- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response

- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Run the scripts to create and configure the DB2 for z/OS database.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the *install\_root\bin\ProfileManagement* directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does

not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Click `Linux_operating_system_menus_to_access_programs > IBM > your_product > Profile Management Tool`.
    - Run the command `install_root/bin/ProfileManagement/pmt.sh`.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:



- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page



When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.

14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
 Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response

- PC\_Adv\_DMgr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMgr\_DB2.response
- PS\_Std\_DMgr\_DB2zOS.response
- PS\_Std\_DMgr\_Oracle.response
- PS\_Std\_DMgr\_SQLServer.response
- PS\_Adv\_DMgr\_DB2.response
- PS\_Adv\_DMgr\_DB2zOS.response
- PS\_Adv\_DMgr\_Oracle.response
- PS\_Adv\_DMgr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses DB2z0S.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_DB2z0S.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
```

```

Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:

```

```

if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root

```

```

signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.

```



```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType

```

```

#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom

```

```

profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:

```

```

Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT
#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses DB2zOS.

```

```

#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####
#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.

```

```

#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:

```

```

true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log

```



```

install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs

```

```

the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage

```

```

Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:

```

```

#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_DB2z0S.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:

```

```

None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####

```

```
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
```



```

Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:

```

```

The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
DB2UDBOS390
Default Values:
DB2UDBOS390
#####
dbType=DB2UDBOS390

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.

```

```

#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

```

```

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied

```

```

and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses DB2zOS.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2zOS.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a

```

```

unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name

```

nodeName=node\_name

```

Parameter: enableAdminSecurity

Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.

Valid Values:
true
Default Values:
true

enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

Parameter: signingCertDN

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

Valid Values:
Varies
Default Values:
None

Parameter: signingCertValidityPeriod

Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

Valid Values:
Varies
Default Values:
None

Parameter: keyStorePassword

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.

Valid Values:
Varies
Default Values:
None

#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

Parameter: enableService
```



```

#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true

```

```

Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:

```

```

Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** **Vista** **Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.

- The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
  11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.

- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.



- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the `manageprofiles` command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the `manageprofiles` command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the `manageprofiles` command-line utility.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir*/profileTemplates/BPM directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response

- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv  
 -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
  - The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:  

```
profile_root\bin\startManager.bat
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):

```
profile_root\bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

**Related information:**

 Add managed node settings (WebSphere Application Server)

*Creating and configuring DB2 for z/OS databases after network deployment profile creation:*

After creating or augmenting profiles, you or your database administrator must create the databases and their tables manually, and you must also run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

*DB2 for z/OS authorization prerequisites:*

Before you create the IBM Business Process Manager databases, you must ensure that the required authorizations are granted for your DB2 for z/OS version. Also ensure that the appropriate storage groups and buffer pools are assigned, and configure clustering if necessary.

### **User authorization requirements for DB2 for z/OS**

Ask your DB2 for z/OS system administrator to check the authorizations that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration. The WebSphere administrator ID should not require more than DBADM authority to define the IBM Business Process Manager databases.

The following storage group, database, and buffer pool GRANT permissions are provided by default, for the WebSphere administrator identified as `#DB_USER#`. These permissions are available in the `createDatabase.sql` database script that is generated by the database design tool:

```
GRANT USE OF STOGROUP #STOGRP# TO #DB_USER# WITH GRANT OPTION;
GRANT DBADM ON DATABASE #DB_NAME# TO #DB_USER#;
GRANT USE OF ALL BUFFERPOOLS TO #DB_USER#;
```

The following GRANT permission might be required to permit the `#DB_USER#` user to create sequences and stored procedures with a schema qualifier of `#SCHEMA#`:

```
GRANT CREATEIN,ALTERIN,DROPIN ON SCHEMA #SCHEMA# TO #DB_USER# WITH GRANT OPTION;
```

The following permissions are also required:

```
GRANT CREATE ON COLLECTION #SCHEMA# TO #DB_USER#;
GRANT BINDADD TO #DB_USER#;
```

### **Authorization requirements for views on DB2 for z/OS V10**

If you are planning to use DB2 for z/OS V10, additional permissions are required for views in the database:

- Before you run the SQL to define views, you might need to set the DBACRVW subsystem parameter to YES.

This setting ensures that WebSphere administrator IDs with DBADM authority on database `#DB_NAME#` can create views for other user IDs.

- On DB2 for z/OS V10, the WebSphere administrator ID must be specifically granted access to views, because access is not implicitly granted to users with DBADM authority on the database. Individual GRANT statements or a Resource Access Control Facility (RACF) group can be used to provide access to views in DB2 for z/OS V10. Ask your DB2 for z/OS administrator to provide this access by using either of the following methods:

- Issue an explicit GRANT statement for each view. For example, the following sample GRANT statements can be issued for user ID WSADMIN:

```
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ACTIVITY_SERVICE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.APPLICATION_COMP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.AUDIT_LOG_B TO WSADMIN WITH GRANT OPTION;
```

```

GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.BUSINESS_CATEGORY_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESCALATION_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.ESC_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.EVENT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.MIGRATION_FRONT TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_ATTRIBUTE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_INSTANCE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPLATE TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.PROCESS_TEMPL_ATTR TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROPERTY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.QUERY_PROP_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.SHARED_WORK_ITEM TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_AUDIT_LOG TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_HISTORY TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_CPROP TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.TASK_TEMPL_DESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_DIST_TARGET TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_BASKET_LDESC TO WSADMIN WITH GRANT OPTION;
GRANT DELETE,INSERT,SELECT,UPDATE ON TABLE S1CELL.WORK_ITEM TO WSADMIN WITH GRANT OPTION;

```

- Define a RACF group that corresponds to the schema name for the views, and connect the WebSphere administrator ID to the RACF group. For example, you can define a RACF group named S1CELL, and connect user WSADMIN to it, as follows:

```

INFORMATION FOR GROUP S1CELL
SUPERIOR GROUP=ZWPS OWNER=ZWPS CREATED=07.144
INSTALLATION DATA=OWNED BY EMP SERIAL 009179, SITE ABCUK
NO MODEL DATA SET
TERMUACC
NO SUBGROUPS
USER(S)= ACCESS= ACCESS COUNT= UNIVERSAL ACCESS=
WSADMIN CONNECT 000000 NONE
CONNECT ATTRIBUTES=NONE
REVOKE DATE=NONE RESUME DATE=NONE

```

### Using the configuration planning spreadsheet to define authorizations

If you have downloaded the configuration planning spreadsheet for use, you can alternatively use this spreadsheet to generate the GRANT permissions that are required for users and for DB2 for z/OS V10 views (as identified in the preceding sections in this topic). The configuration planning spreadsheet is available from Techdoc WP102075 in the IBM Support Portal.

The Database worksheet in the spreadsheet lists a set of sample SQL statements that can be used to create the databases and storage groups. Additionally, the GRANT permissions that are required to authorize the WebSphere administrator and to provide access to DB2 for z/OS V10 database tables are provided. When you specify the user and database object names on the BPMVariables worksheet of the spreadsheet, these values are propagated to the Database worksheet, and are used to complete the CREATE and GRANT statements with the appropriate values.

Ask your DB2 for z/OS system administrator to use the relevant CREATE statements to create the databases and storage groups, and to use the GRANT statements to authorize the WebSphere administrator. For more information about using the artifacts generated from the spreadsheet, see the accompanying PDF document in the Techdoc.



## Storage group assignments and buffer pool usage

Ask your DB2 for z/OS system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log, but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over for use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.

## Clustering requirements

If you intend to configure clustering, your DB2 for z/OS system must be running in data-sharing mode.

### Related tasks:

“Creating database design files and database scripts (AIX stand-alone)” on page 1123

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (AIX network deployment)” on page 1413

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux stand-alone)” on page 273

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Linux network deployment)” on page 565

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris stand-alone)” on page 1968

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Solaris network deployment)” on page 2260

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating database design files and database scripts (Windows stand-alone)” on page 2832

Use the database design tool to generate a design file that is used when creating a stand-alone profile. The database design tool also generates database scripts that you can use to create your database tables.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

### *Creating databases in the DB2 for z/OS subsystem:*

When you use the database design tool (**DbDesignGenerator**) to define your database configuration, you can generate the database scripts that are required for creating the databases for the IBM Business Process Manager components.

You can use various tools to run these database scripts:

- When you run the database design tool to generate the database scripts, the **createDB2.sh** script is also generated. You can use **createDB2.sh** to run the database scripts.



- You can use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to run the database scripts.

### Choosing which tool to use

You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 for z/OS objects, particularly in a production environment.

#### Considerations for choosing the `createDB2.sh` script

- **createDB2.sh** can create all your database objects in one simple execution of the tool, so it is a good choice if this is your first server implementation.
- **createDB2.sh** runs the database scripts that the database design tool (**DbDesignGenerator**) generates.
- **createDB2.sh** runs the SQL for each component in the correct sequence.
- **createDB2.sh** creates database objects according to a naming convention that you define.
- **createDB2.sh** organizes the layout of database objects across DB2 for z/OS databases.
- **createDB2.sh** issues GRANT permissions to database, storage group, and buffer pool objects.
- **createDB2.sh** automatically completes customization of the service integration bus Data Definition Language (DDL).
- **createDB2.sh** runs in a UNIX System Services environment.
- **createDB2.sh** produces an audit trail of the objects that it creates.

#### Considerations for choosing other tools

- You might prefer to use the DB2 command line processor to run the SQL statements in the UNIX Systems Services environment. You must first run the **createDB2.sh** script in a mode that bypasses execution of the SQL statements and instead consolidates them into two `.sql` files. You can then run these files by using the command line processor.
- There is no restriction on the naming or organization conventions that apply to the database objects other than the standard database subsystem restrictions.
- Some tools can be run from a z/OS environment.
- The tools can produce an audit trail of the DB2 database commands that have been issued.

#### Related tasks:

“Creating database design files and database scripts (Windows network deployment)” on page 3130  
 You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

#### *Configuring the DB2 command line processor:*

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

Ensure that a properties file, for example, `clp.properties`, exists for the DB2 command line processor. If required, you can create your own properties file by using the sample properties file that is available in the directory where the command line processor is installed. For more information, see your DB2 for z/OS documentation.

Complete the following configuration steps in the z/OS UNIX System Services environment from which the **createDB2.sh** script will be run:

1. Configure the DB2 command line processor for each user ID that will work with DB2 for z/OS from the command line. You can update the personal or shared user profiles as follows:

- Modify the CLASSPATH environment variable to include the clp.jar file.
- Use the CLPPROPERTIESFILE environment variable to define the fully qualified name of the properties file for the command line processor.
- Define the **db2** command as an alias for the command that starts the command line processor.
- Specify the DB2JccConfiguration.properties file that defines the JDBC properties to be applied to the command line processor.

You can use the following syntax to add the required entries to the .profile or /etc/profile file:

```
export CLPHOME=clp_install_dir
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=clp_properties_file_path
alias db2="java -Ddb2.jcc.propertiesFile=
/file_path/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

For example:

```
export CLPHOME=/shared/db2910_base
export CLASSPATH=$CLASSPATH:$CLPHOME/lib/clp.jar
export CLPPROPERTIESFILE=/wasv8config/clp.properties
alias db2="java -Ddb2.jcc.propertiesFile=
/wasv8config/DB2JccConfiguration.properties com.ibm.db2.clp.db2"
```

2. In the properties file for the command line processor, define alias names that can be used to connect to the DB2 for z/OS server. An alias name definition can include the following entities:
  - A URL that specifies the domain name or IP address of the database server, the port on which the server listens, and the DB2 location name defined during installation. The URL can take the form: *server:port/database*. The port is optional, and the DB2 location name must be specified in uppercase characters.
  - A user ID and an associated password that can be used to connect to the DB2 server. This user ID should correspond to the user ID (with SYSADM authority) that the DB2 system administrator uses to run the **createDB2.sh** script.

You can add the required alias name entries to the properties file by using the following syntax:

```
DB2ALIASNAME=URL,user_ID,password
```

For example:

```
DSNXWBD=localhost:9446/DSNXWBD,SYSADM1,SYSPWRD1
```

**Tip:** When you define a *DB2ALIASNAME* value in the properties file, ensure that the correct connection details are specified to avoid connecting to the wrong database and inadvertently overwriting its contents.

3. Configure the DB2 DBACRVW subsystem parameter to enable user IDs with DBADM authority on a database to perform the following tasks for other user IDs: create views on tables in the database, create aliases for tables, and create materialized query tables. You can use the installation Command List (CLIST) to access the DSN TIPSPF panel and update the DBADM CREATE AUTH field to set DB2 ZPARAM DBACRVW=YES.

Use the **createDB2.sh** script to either run the SQL to create the database objects for a specified database, or to consolidate the SQL statements into two .sql files that you can run later using other database tools.

#### Related tasks:

“Creating DB2 for z/OS database objects using the createDB2.sh script” on page 3368

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 3370

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (z\_schema.sql and z\_schemaProc.sql) that you can run later by using the DB2

command line processor.

*Creating DB2 for z/OS database objects using the createDB2.sh script:*

After creating the database scripts for the IBM Business Process Manager components, you can run the **createDB2.sh** script to create one or more databases in the DB2 for z/OS subsystem and populate each database with objects.

You can also use tools such as the DB2 command line processor, SPUFI, or DSNTEP2 to create and populate databases.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

The number of databases that you create for your IBM Business Process Manager installation can depend on your topology, and how you want to distribute your database objects across the DB2 for z/OS subsystem. Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created. For example:

- To create a single database, you can run **createDB2.sh** once from a directory that contains all the database scripts that were generated for all the IBM Business Process Manager components.
- To create multiple databases at the cell level and cluster levels, you can run **createDB2.sh** once from each directory that contains the database scripts required to create the database objects for the cell and clusters.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:  

```
cd /u/work/S4CELLDB
```
2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCcreate
-DBVCat volume_catalog -DBUser DB_user_ID -RunSQL
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCcreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**-RunSQL**

Runs the SQL statements that create the database objects.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCcreate -DBVCat DSNV10PP
-DBUser S4ADMIN -RunSQL
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

3. Review the messages that are displayed in the console, checking that no error messages are displayed.

**Tip:** The first time that you run **createDB2.sh** to create the database, you see a few messages because the script first attempts to drop the database, which at that stage does not yet exist. These messages can be ignored, and subsequent invocations of **createDB2.sh** for the same database will not produce such messages.

When the script has finished running, you can also review the `z_output.txt` file, which provides an audit trail of the operations completed and status messages. This file is stored in the directory from which you ran the **createDB2.sh** script.

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 3366

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.

“Creating database design files and database scripts (Windows network deployment)” on page 3130

You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using the DB2 command line processor:*

When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

You can also run these `.sql` files by using any other database tool of your choice, such as SPUIFI or DSNTEP2.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a network deployment environment, you might define two DB2 for z/OS databases, where one database is used for the Common database component, which is cell-scoped, and another database is used for all the other components, which are cluster-scoped.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, including the **createDB2.sh** script, which are required to create the database, into that directory.
- Use FTP to transfer the database scripts, including the **createDB2.sh** script, to the z/OS system that contains the DB2 for z/OS installation. Transfer the **createDB2.sh** script as an ASCII text file, and transfer the database schema files in binary mode. Also ensure that you retain the directory structure when you transfer the files.
- Grant execute permissions to the **createDB2.sh** script.
- Create or augment the profile.
- Configure the DB2 command line processor.

Provided that the database scripts are grouped appropriately in the output directories that were specified when you ran the database design tool, you can run the **createDB2.sh** script once for each instance of a database to be created.

**Important:** It is anticipated that the DB2 for z/OS system administrator will run the **createDB2.sh** script with a user ID that has SYSADM authority, to create the databases and storage groups. On completion, the system administrator can then grant DBADM authority for the IBM Business Process Manager databases to the WebSphere administrator.

Complete the following steps for each database that you want to create:

1. In the z/OS system that contains the DB2 installation, access the UNIX System Services command shell and then change to the directory to which you transferred the database scripts for the database objects to be created. For example:

```
cd /u/work/S4CELLDB
```

2. Run the **createDB2.sh** script by using the following syntax:

```
createDB2.sh -DBAlias alias_name -DBName database_name -DBSto storage_group_name -DBCreate
-DBVCat volume_catalog -DBUser DB_user_ID
```

where:

**-DBAlias**

Specifies an alias name, which maps to the DB2 server URL, user ID, and password, and which is used to connect to DB2. If you do not specify this parameter when you run the **createDB2.sh** script, you are prompted for a value.

**-DBName**

Specifies the name of the database to be created.

**-DBSto**

Specifies the storage group name for the volumes that hold the data sets where tables and indexes are stored.

**-DBCreate**

Creates the database.

**-DBVCat**

Specifies the name of the DB2 Virtual Storage Access Method (VSAM) catalog that records information about the data sets.

**-DBUser**

Specifies the user ID of the WebSphere administrator that has database administration authority for the IBM Business Process Manager database being created.

**Tip:** By omitting the **-RunSQL** parameter, you bypass the execution of the SQL statements that create the database objects, and instead consolidate the statements into two `.sql` files.

For example:

```
createDB2.sh -DBAlias DSNXWBD -DBName S4CELLDB -DBSto S4DBSTO -DBCreate -DBVCat DSNV10PP
-DBUser S4ADMIN
```

For information about all the parameters that are available for the **createDB2.sh** script and example usage, see `createDB2.sh` script.

The `z_schema.sql` and `z_schemaProc.sql` files, which hold the consolidated SQL statements, are generated in the directory from which you ran the **createDB2.sh** script. The `z_schemaProc.sql` file contains SQL statements for stored procedures, and is generated in addition to `z_schema.sql` because stored procedures require the “at” sign (@) as a statement termination character when running the command line processor.

3. Run the DB2 command line processor in batch mode to run the SQL commands in the `z_schema.sql` file:

```
db2 -f /createDB2.sh_path/z_schema.sql
```

For example:

```
db2 -f /u/work/S4CELLDB/z_schema.sql
```

The command line processor reads the data in the file, and processes the commands in the file consecutively.

4. Run the DB2 command line processor again to run the SQL commands in the `z_schemaProc.sql` file, and additionally use the **-td** parameter to define @ as the statement termination character:

```
db2 -td@ -f /createDB2.sh_path/z_schemaProc.sql
```

For example:

```
db2 -td@ -f /u/work/S4CELLDB/z_schemaProc.sql
```

Each database is created and populated with the required database objects.

**Related tasks:**

“Configuring the DB2 command line processor” on page 3366

Before you run the **createDB2.sh** script, you must configure the DB2 command line processor by defining a set of environment variables and a **db2** command alias. You must also define alias names that can be used to connect to the DB2 for z/OS server.



“Creating database design files and database scripts (Windows network deployment)” on page 3130  
You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

*Creating DB2 for z/OS database objects using SPUFI or DSNTEP2:*

You can use tools such as SPUFI or DSNTEP2 to run the database scripts that are used to create the DB2 for z/OS database objects for your configuration.

- Decide on the DB2 for z/OS database design and configuration, including the number of databases required for the IBM Business Process Manager components. For example, in a stand-alone server environment, you might decide to set up all the component databases in a single DB2 for z/OS database.
- Decide on a naming convention for the DB2 for z/OS objects, such as databases, storage groups, VSAM catalog names (VCATs), schema prefixes, buffer pools, and security IDs.
- Create the required buffer pools. For more information, see Sample DB2 commands and SQL.
- Run the database design tool (**DbDesignGenerator**) to generate the database scripts that can be used to create the database objects for the IBM Business Process Manager components. Set up your database configuration so that the database design tool creates a single output directory for each database to be created, and generates all the relevant database scripts, which are required to create the database, into that directory.

**Tip:** The generated scripts that you use to create the database objects can either be in ASCII or EBCDIC format. Typically, the generated .sql files are in ASCII format, and the equivalent .ddl files are in EBCDIC format.

- Use FTP to transfer the database scripts to the z/OS system that contains the DB2 for z/OS installation.
- Create or augment the profile.

You can create the database objects by using the tool of your choice. For example:

**SPUFI** A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

#### **DSNTEP2**

A sample dynamic SQL program provided with the DB2 for z/OS product.

1. Assign the appropriate read permissions to the SQL scripts; for example:  

```
chmod 644 createTable_AppScheduler.sql
```
2. If the tool that you want to use to view, edit, and run the scripts requires the scripts to be in EBCDIC format, rather than ASCII format, use the **iconv** command to convert the files to EBCDIC. For example:  

```
iconv -t IBM-1047 -f IS08859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

**Important:** After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed-width MVS data sets.

**Tip:** If you have converted the files from ASCII format to EBCDIC, but need to run the files in ASCII format, you can also use **iconv** to convert the files back to ASCII. For example:

```
iconv -t IS08859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

3. Edit the values in the createDatabase.sql and dropDatabase.sql files according to your requirements. The following symbolic variables need to be updated:



@STOGRP@ = DB2 Storage group name  
 @DB\_NAME@ = DB2 Database name  
 @SCHEMA@ = DB2 SQLID (Schema Qualifier)  
 @BP4K@ = Buffer Pool of 4k Size  
 @BPINDEX@ = Buffer Pool Index  
 @VOLUMES@ = DB2 comma separated list of volume-ids  
 @VCAT@ = DB2 integrated catalog facility name  
 @DB\_USER@ = The user ID which will be used to create the database objects

4. To create database objects outside of the z/OS UNIX environment by using SPUFI or DSNTEP2, copy the customized database scripts from z/OS UNIX to a partitioned data set.
5. Run the customized scripts by using the tool of your choice.

**Tip:** If you previously ran the **createDB2.sh** script and chose to bypass execution of the SQL scripts, you can also run the consolidated SQL files (`z_schema.sql` and `z_schemaProc.sql`), which were generated.

6. Verify that the database tables are created successfully with no errors by inspecting the output.

**Related tasks:**

“Creating database design files and database scripts (Windows network deployment)” on page 3130  
 You can specify the design of your database configuration by using the database design tool. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

“Creating DB2 for z/OS database objects using the DB2 command line processor” on page 3370  
 When you run the **createDB2.sh** script, you can choose to run the SQL immediately to create the database objects for the specified database, or you can bypass running the SQL scripts so that you can run them later. If you bypass execution of the SQL, the **createDB2.sh** script consolidates the SQL statements into two files (`z_schema.sql` and `z_schemaProc.sql`) that you can run later by using the DB2 command line processor.

*Creating the messaging engine data stores:*

If the messaging engine data stores have not already been created, use the database design tool to generate the database scripts that the database administrator will use to create the tables for the messaging engine data stores.

Before you create and run the database scripts:

- Create a cluster.
- Decide which service integration bus components you want to create in your network deployment configuration.

You must create and populate a messaging engine data source for each of the following components that you configure:

- Service Component Architecture system bus
- Service Component Architecture application bus
- Process Server
- Performance Data Warehouse
- Business Process Choreographer
- Common Event Infrastructure

You can create all the database objects in one or more databases in the DB2 for z/OS subsystem. Take note of the following considerations:

- All messaging engines use the same table names. Therefore, the tables for each messaging engine require a unique schema qualifier.

- When you use the database design tool to define the database configuration for each messaging engine, you can specify a prefix for table space names:
    - If you intend to use a single database for all the messaging engines, specify a unique table space prefix for each messaging engine to make its table space name unique within the database.
    - If you intend to use a separate database for each messaging engine, the table space prefixes that you specify for the messaging engines can be identical or unique.
1. For each messaging engine, complete the following steps to generate the required database scripts:
    - a. Run the database design tool (**DbDesignGenerator**) from the *install\_root/util/dbUtils* directory.
    - b. Select option **(2)Create a database design for a single component**, and then select the appropriate option for the messaging engine.
    - c. Complete the remaining prompts in the tool to define the database configuration and to generate the scripts.

**Tip:** As an alternative to running the database design tool multiple times to generate scripts for each of the messaging engines, you can run the tool once using option **(1)Create a database design for Standalone profile or Deployment Environment**. With this option, you can define the database configuration and generate database scripts for the following components in a single pass through the tool: Common database, Business Process Choreographer, Performance Data Warehouse, Process Server, Business Space, and the six messaging engines.

2. Run the generated DDL scripts using the tool of your choice.

The messaging engine data stores are created.

**Related tasks:**

“Creating database design files and database scripts for specific components with DB2 for z/OS in a network deployment environment” on page 3136

You can use the database design tool to generate a design file, and optionally database scripts, for the database tables required by specific IBM Business Process Manager components.

“Creating database design files and database scripts for deployment environments with DB2 for z/OS” on page 3130

Use the database design tool to generate a design file that is used to configure your databases when creating a deployment manager profile, or when using the deployment environment wizard. The database design tool also generates database scripts that you can use to create your database tables.

*Granting table privileges to the JCA authentication alias user ID:*

If the schema name you are using is not the same as the JCA authentication alias user ID, you must grant a subset of DB2 for z/OS privileges to the JCA authentication alias user ID.

The database scripts for the service integration bus (SIB) contain commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply GRANT statements.

Use a schema name that is different from the JCA authentication alias to prevent the alias user ID from having the authority to drop tables. (The authority to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to drop tables.

If you want IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the database scripts and editing them to construct GRANT commands from the CREATE commands. You can create GRANT commands like the one shown in the following example:

```
GRANT ALL PRIVILEGES ON TABLE
cell.tablename TO userid/sqlid
```

where *userid/sqlid* is the JCA authentication alias user ID.

**Note:** Typically, the creator of a database object has implicit use of that object without requiring additional GRANT permissions. However, for DB2 for z/OS Version 10, additional GRANT permissions might be required for views because access to views is not implicitly granted to the creator.

*Setting the correct schema name for the messaging engines:*

To ensure that the service integration bus (SIB) messaging engines can access the appropriate DB2 for z/OS tables, the correct schema name must be set for the messaging engines. You can use the administrative console to change the schema names.

Start the server.

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each messaging engine:
  - a. Select **Messaging engines**, and then click the name that is displayed.
  - b. Click **Message store**.
  - c. Change the value of **Schema name** to the name used when creating the DB2 for z/OS tables for this messaging engine.
  - d. Click **Apply**.
  - e. Save your configuration changes.
4. Verify that the messaging engines can be started:
  - a. Log out of the administrative console.
  - b. Stop, and then restart the server.
  - c. Look in the output of the server log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

*Creating or augmenting network deployment profiles with an Oracle database server:*

You can configure a network deployment environment for IBM Business Process Manager using an Oracle database server.

*Preparing the Oracle database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

To configure the Common database while creating or augmenting the profile, you must know the service name for your Oracle instance.

1. Create a table space.

```
SQL> CREATE TABLESPACE [tablespace_name] DATAFILE '[datafile_name]' SIZE 50M AUTOEXTEND ON NEXT 10M
MAXSIZE UNLIMITED LOGGING;
```
2. Create the users. You will specify these users on the Database Configuration - Part 2 page in the Profile Management Tool. The following example assumes you create a BPMDBA user name for the Process Server database, a PDWDBA user name for the Performance Data Warehouse database, and a CMNDBA user name for the Common database:

```
SQL> CREATE USER BPMDBA IDENTIFIED BY [password] DEFAULT TABLESPACE [tablespace_name];
SQL> CREATE USER PDWDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
SQL> CREATE USER CMNDBA IDENTIFIED BY [password] DEFAULT TABLESPACE
[tablespace_name];
```

3. Grant privileges to the users that you specified in the previous step. For example:

```
SQL> GRANT ALL PRIVILEGES TO BPMDBA;
SQL> GRANT ALL PRIVILEGES TO PDWDBA;
SQL> GRANT ALL PRIVILEGES TO CMNDBA;
```

4. Grant execute privileges on DBMS\_LOCK. For example:

```
SQL> GRANT execute on DBMS_LOCK to BPMDBA;
SQL> GRANT execute on DBMS_LOCK to PDWDBA;
SQL> GRANT execute on DBMS_LOCK to CMNDBA;
```

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.



12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.



- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 207. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.

20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
  - Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  - Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
  - Configure Business Process Rules Manager.
1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.

- Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.
  9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
  10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
    - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.

- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

`profile_root\properties\portdef.prop`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.



- c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 208. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_Standalone\_DB2.response
- PC\_Std\_Standalone\_DB2z0S.response
- PC\_Std\_Standalone\_Oracle.response
- PC\_Std\_Standalone\_SQLServer.response
- PC\_Adv\_Standalone\_DB2.response
- PC\_Adv\_Standalone\_DB2z0S.response
- PC\_Adv\_Standalone\_Oracle.response
- PC\_Adv\_Standalone\_SQLServer.response



- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId db7user -dbPassword db7secret
-dbType ORACLE -procSvrDbName BPMDB -perfDWDbName PDWDB
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

Response file for creating deployment manager profiles:

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

Sample response file for an Advanced Process Center deployment manager profile:

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
```

```

When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.

```

```

Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile

```

```

parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you

```

```

as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign

```



```

#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath

```

```

#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:

```

```

The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be

```

```

accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

```

```

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMgr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties

```

```

file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword

```

```

along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.

```



```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#

```

```

Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winServiceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceCheck
#
Description:

```

```

The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing

```

```

environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses Oracle.
#

```

```

Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#

```

```

Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true

```

```

Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile. Do not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:

```



```

None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:

```

```

None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.

```

```

#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbldbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false
#####

```

```

Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: dbLocation
#
Description:
The directory of the ORACLE_HOME directory. This parameter is required
when the parameter dbDelayConfig is set to false. No, this parameter is
only valid if you are using Oracle databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbLocation=db_location
#####
Parameter: dbDelayConfig

```

```

#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd

#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.

```

```

#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#

```

```

Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMGr_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.

```



```

Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr
#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01
#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName

```

```

#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

```

```

#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.

```

```

During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName

```

```

#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

```

```

#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true

```

```
Default Values:
false
#####
#isDeveloperServer=false
```

*Augmenting deployment manager profiles for Process Center with an Oracle database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.



1. Use one of the following methods to start the Profile Management Tool.
  - Start the tool from the First steps console.
  - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
  - Run the command `install_root\bin\ProfileManagement\pmt.bat`
2. Shut down any servers associated with a profile you plan to augment.
3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. **Advanced:** On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
  - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.  
Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.

- a. Select **Use a database design file**.
- b. Click **Browse**.
- c. Specify the fully qualified path name for the design file.
- d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 209. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with an Oracle database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.



- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. **Advanced:** On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. **Advanced:** On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. **Optional:** Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system. When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.



- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the

service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Oracle**.
  - b. Optional: If you want to change the default values, enter new names for the required databases.
    - The database names can be the same.
    - You must input the database service name for each database.
    - The database must already exist.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. You cannot create a new database using Oracle.

**Important:** You must have a user ID that has SYSDBA privileges before creating any profile. Specify the following fields.

Table 210. Required database configuration fields for Oracle - Part 2

Field	Action needed
Common database user name	Enter the user name to authenticate with the database.
Common database password	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Location (directory) of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
Database server host name (for example IP address)	Accept the default value of localhost or enter the correct database server host name.

Table 210. Required database configuration fields for Oracle - Part 2 (continued)

Field	Action needed
Server port	Accept the default value of 1433 or enter the correct server port number.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.

3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- `PC_Std_DMGr_DB2.response`
- `PC_Std_DMGr_DB2z0S.response`
- `PC_Std_DMGr_Oracle.response`
- `PC_Std_DMGr_SQLServer.response`
- `PC_Adv_DMGr_DB2.response`
- `PC_Adv_DMGr_DB2z0S.response`
- `PC_Adv_DMGr_Oracle.response`
- `PC_Adv_DMGr_SQLServer.response`
- `PC_Std_Managed_DB2.response`
- `PC_Std_Managed_DB2z0S.response`
- `PC_Std_Managed_Oracle.response`
- `PC_Std_Managed_SQLServer.response`
- `PC_Adv_Managed_DB2.response`
- `PC_Adv_Managed_DB2z0S.response`
- `PC_Adv_Managed_Oracle.response`
- `PC_Adv_Managed_SQLServer.response`
- `PC_Std_StandAlone_DB2.response`
- `PC_Std_StandAlone_DB2z0S.response`
- `PC_Std_StandAlone_Oracle.response`
- `PC_Std_StandAlone_SQLServer.response`
- `PC_Adv_StandAlone_DB2.response`
- `PC_Adv_StandAlone_DB2z0S.response`
- `PC_Adv_StandAlone_Oracle.response`
- `PC_Adv_StandAlone_SQLServer.response`
- `PS_Std_DMGr_DB2.response`
- `PS_Std_DMGr_DB2z0S.response`
- `PS_Std_DMGr_Oracle.response`
- `PS_Std_DMGr_SQLServer.response`
- `PS_Adv_DMGr_DB2.response`
- `PS_Adv_DMGr_DB2z0S.response`

- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

*Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
  1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.



- c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
  - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

**Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

**Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.



Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
  - To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available for custom profiles:

- managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example 'personalCertValidityPeriod=1 ' or 'winserviceCheck=false '. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response

- PS\_Std\_Standalone\_DB2.response
- PS\_Std\_Standalone\_DB2zOS.response
- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT
```

```
#####
```

```

These response file has the applicable parameters for creating a
Managed PC Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this

```

```

parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.

```

```

#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An

```



```

INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#

```

```

Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort

```

```

parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####

```

```

Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####

```

create

```

Parameter: templatePath

Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.

When using the -templatePath parameter, specify the fully qualified file
path for the parameter.

Valid Values:
Varies
Default Values:
None

templatePath=BPM/managed.procctr
```

```

Parameter: profileName

Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.

Valid Values:
Varies
Default Values:
None

profileName=Custom01
```

```

Parameter: cellName

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
```

```

where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:

```



```

Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual

```

```

automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN
#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.

```

```

This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.

```

```

Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT
#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_Oracle.response

```

```

#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####

```

profileName=Custom01

```

Parameter: cellName

Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

Parameter: nodeName

Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

Valid Values:
Varies
Default Values:
Varies

cellName=cell_name
nodeName=node_name

Parameter: enableAdminSecurity

Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.

Valid Values:
true
Default Values:
true

enableAdminSecurity=true
adminUserName=adminUser_ID
```



adminPassword=adminPassword

```

Parameter: signingCertDN

Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.

Valid Values:
Varies
Default Values:
None

Parameter: signingCertValidityPeriod

Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.

Valid Values:
Varies
Default Values:
None

Parameter: keyStorePassword

Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.

Valid Values:
Varies
Default Values:
None

#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

Parameter: enableService

Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.

Valid Values:
false
```

```

true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify

```

```

false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
ORACLE
Default Values:
ORACLE
#
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbJDBCClasspath

```

```

#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM

```

```

file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses Oracle.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_Oracle.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various

```

```

directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```



```

Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the

```

```

root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the

```

```

user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled

```

```

Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbDriverType
#
Description:
The database driver type. For an Oracle database, specify ORACLE. For
databases other than Oracle, the value is automatically set based on the
server operating system. Server installations on all operating systems use
type 4.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbDriverType=ORACLE_THIN

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the

```

```

manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage

```

```

Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have

access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.



When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

#### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

#### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### Related information:

 Updating ports in existing profiles (WebSphere Application Server)

*Augmenting custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.

- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response

- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.

- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:
 

```
profile_root\bin\startManager.bat
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):
 

```
profile_root\bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### Related information:

 [Add managed node settings \(WebSphere Application Server\)](#)

*Creating or augmenting network deployment profiles with an SQL Server database server:*

You can configure a network deployment environment for IBM Business Process Manager using a Microsoft SQL Server database server.

*Preparing the SQL Server database server:*

During the profile creation or augmentation process, you can configure the Common database that is used by selected components, or you can postpone the database configuration by producing scripts that must be run manually by you or your database administrator (DBA). The other required databases are configured during network deployment.

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

**Restriction:** The Process Server database (BPMDB) and the Performance Data Warehouse database (PDWDB) must NOT be case-sensitive. Other databases can be case-sensitive. If you are using the SQL files to create the database for Business Process Choreographer, the SQL files create case-sensitive names for the databases.



When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

The following table shows the database privileges that are required to access the data store.

Table 211.

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ\_COMMITTED. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is\_read\_committed\_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ\_COMMITTED ON**.

*Configuring XA transactions:*

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server. The SQL Server JDBC driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error when the server starts: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'".**

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

**On Windows XP and Windows Server 2003:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** window, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

**On Windows Vista and Windows 7:**

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** window.
- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
- f. Click **OK** again to close the Properties window, and then close Component Services.



- g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
  - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from the Microsoft Site using the URL from Resources section.
  - b. Extract the archive to any folder.
  - c. Copy the sqljdbc\_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc\_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc\_xa.dll file in the x64 folder.
  - d. Run the xa\_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc\_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
  - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in the master database (for example, for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

*Creating or augmenting deployment manager profiles:*

To start the network deployment configuration, create or augment a deployment manager. You can create profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

*Creating Process Center deployment manager profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Center using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a

com.ibm.wsspi.profile.WSProfileException or Access is denied message occurs in the *install\_root\bin\ProfileManagement\pmt.bat* file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command *install\_root\bin\ProfileManagement\pmt.bat*
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is *install\_root\profiles\profile\_name*.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.

- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root/properties/portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process

Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 212. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
<b>Common database user name</b>	Enter the user name to authenticate with the database.
<b>Common database password</b>	Enter a password to authenticate with the database.
<b>Confirm password</b>	Confirm the password.
<b>Location (directory) of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.



18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating Process Server deployment manager profiles with an SQL Server database server using the Profile Management Tool:*

You can configure a deployment manager profile for Process Server using the Profile Management Tool.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).

- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  3. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  4. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  5. On the Profile Creation Options page, choose to perform a **Typical** or an **Advanced** profile creation, and click **Next**. If you selected **Typical** profile creation, skip to the Administrative Security step.
  6. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
    - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
  8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
    - In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
    - In the **Server name** field, enter a name for the server or accept the default value.
    - In the **Host name** field, enter a name for the host or accept the default value.
    - In the **Cell name** field, enter a name for the cell or accept the default value.
 Click **Next**.



9. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values. Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.
10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do

not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. Advanced: If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

14. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

15. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes.
  - d. Click **Next**.
16. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 213. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
<b>Common database user name</b>	Enter the user name to authenticate with the database.
<b>Common database password</b>	Enter a password to authenticate with the database.
<b>Confirm password</b>	Confirm the password.
<b>Location (directory) of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.

17. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
18. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
19. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
20. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Creating deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create new profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

- dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
  3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example '**personalCertValidityPeriod=1** ' or '**winserviceCheck=false** '. Spaces will cause profile creation to fail.

The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2zOS.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2zOS.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2zOS.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response

- PC\_Adv\_Managed\_DB2zOS.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2zOS.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2zOS.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2zOS.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating deployment manager profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbType MSSQLSERVER_MICROSOFT
-dbHostName dbHost.com -procSvrDbName BPMDB -dbProcSvrUserId procadmin -dbProcSvrPassword procpwd
-perfDWDName PERFDB -dbPerfDWUserId perfadmin -dbPerfDWPassword perfpwd -dbName CMNDB
-dbCommonUserId commonadmin -dbCommonPassword commonpwd
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

The following example commands are used to create a deployment manager profile using the **manageprofiles** command that includes the following components: CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space.

```
install_root\bin\manageprofiles -create -templatePath install_root\profileTemplates\BPM\
\dmgr.procsvr.adv -dbHostName hostName -dbServerPort 1433 -dbDelayConfig true
-configureBspace true -dbType MSSQLSERVER_Microsoft -dbUserId userID -dbJDBCClasspath
path to JDBC 2.0 driver -dbName CMNDB
-dbPassword password
```

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

*Response file for creating deployment manager profiles:*

You can use the response file provided here with the **manageprofiles** command to create a deployment manager profile.

*Sample response file for an Advanced Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
```



```

#
was.install.root/bin/manageprofiles -response PC_Adv_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:

```



```

None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####

```

```
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
```

```
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
```

```

#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName

```

```

#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false

```

```

true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbiddbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file

#####
Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:

```

```

The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer
#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false
#####
Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.

```

```

Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false

```



```

true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procctr

#####
Parameter: profileName
#
Description:

```

```

Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.

```

```

#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

#####
Parameter: enableService

```

```

#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####

```



```

#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM

```

```

5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a

```

```

trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:

```

```

Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

```

```

#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.
Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the

```

```

manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#

```

```

Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual
#####
Parameter: bpmdbDesign
#
Description:
Specifies the file that holds the database configuration design for all of
the IBM Business Process Manager components .
This parameter replaces the -wbidbDesign that was used in previous
versions of WebSphere Process Server.
#
Note: If a bpmdbDesign file parameter is specified, none of the database
related properties should be specified.
#
Valid Values:
Varies
Default Values:
None
#####
#bpmdbDesign=design_file
#####

```



```

Parameter: dbCreateNew
#
Description:
Indicates a new database is created or if an existing database is to be
reused. Valid values are true or false. This value cannot be true for an
Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
a DB2 database.
#
Valid Values:
false
Default Values:
false
#####
dbCreateNew=false

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbDelayConfig
#
Description:
Indicates whether to postpone table creation until after the profile is
created. Valid values are true or false. The parameter is set to false by
default. Setting this parameter to true delays the execution of the
scripts that create the database tables. No. Specify false if you want the
database to be set up during profile creation.
#
Valid Values:
false
true
Default Values:
false
#####
dbDelayConfig=false

#####

```

```

Parameter: dbName
#
Description:
The name of the database. By default, the value is set to orcl for Oracle
databases and to CMNDB for all other supported databases.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
CMNDB
#
Parameter: dbUserId
#
Description:
The user ID for all database types. Specifies the user ID that has
privileges to create and drop the databases. The WebSphere data source
uses this ID to authenticate the database connection.
Important: The -dbUserId parameter value must be a valid database
authorization ID. Required if the -bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbPassword
#
Description:
The password required for database authentication. Required when the
-bpmdbDesign file is not set.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd
#####
Parameter: dbCommonUserId
#
Description:
The user id to authenticate with the Common database.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dbCommonPassword
#
Description:
The password for the user id of the dbCommonUserId parameter.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:

```

```

Varies
Default Values:
None
#####
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer

```

```

people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server deployment manager profile:*

Modify and use this response file to create a deployment manager profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
DMgr PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_DMgr_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with

```

```

the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/dmgr.procsvr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Dmgr01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (Dmgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies

```

```

#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter

```

```

with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: defaultPorts
#
Description:
Assigns the default or base port values to the profile.d o not use this
parameter when using the -startingPort or -portsFile parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: startingPort
#
Description:
Specifies the starting port number for generating and assigning all ports
for the profile.
Port values are assigned sequentially from the -startingPort value,
omitting those ports that are already in use. The system recognizes and
resolves ports that are currently in use and determines the port
assignments to avoid port conflicts.
Do not use this parameter with the -defaultPorts or -portsFile parameters.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: portsFile
#
Description:
An optional parameter that specifies the path to a file that defines port
settings for the new profile.

```



```

Do not use this parameter when using the -startingPort or -defaultPorts
parameter.
During profile creation, the manageprofiles command-line utility uses an
automatically generated set of recommended ports if you do not specify the
-startingPort parameter, the -defaultPorts parameter or the -portsFile
parameter. The recommended port values can be different than the default
port values based on the availability of the default ports.
#
Valid Values:
Varies
Default Values:
None
#####
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem

```

```

#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password

```

```

#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false

```

```
true
Default Values:
false
#####
#isDeveloperServer=false
```

*Augmenting deployment manager profiles for Process Center with an SQL Server database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.

- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.
    - d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.

- e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including `tw_admin` and `tw_user`. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server as a Windows service and issue the **startServer** command, the **wasservice** command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process



Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 214. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
<b>Common database user name</b>	Enter the user name to authenticate with the database.
<b>Common database password</b>	Enter a password to authenticate with the database.
<b>Confirm password</b>	Confirm the password.
<b>Location (directory) of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.

20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles for Process Server with an SQL Server database server using the Profile Management Tool:*

You can use the Profile Management Tool to augment an existing WebSphere Application Server V8.0 deployment manager profile.

Remember to shut down any servers associated with a profile you plan to augment.

If you want to configure the Common database while creating the profile, make sure that your database server is installed and running.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

#### Restrictions:

Vista

Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Specify a database design file to be used for the database configuration.

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
- Choose to configure IBM Forms Server to work with Human Task Management widgets in Business Space.
- Configure Business Process Rules Manager.
  1. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  2. Shut down any servers associated with a profile you plan to augment.
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  5. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  6. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
7. If you selected **Typical** profile creation, skip to the Administrative Security step.
  8. Advanced: On the Optional Application Deployment page, select whether to deploy the administrative console for managing the server (recommended). Click **Next**.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system. When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. Required: On the Administrative Security page, enter values for the **User name**, **Password**, and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw\_admin and tw\_user. Because all IBM Business Process Manager profiles must have administrative security enabled, **Next** is enabled only after you enter the values.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration - Part 1 step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.

- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the `key.p12` file or the `root-key.p12` file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. **Advanced:** On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the `updatePorts.ant` file through the `ws_ant` script detailed in this topic.

15. **Advanced:** If you do not have administrative privileges, skip to the next step. If you have administrative privileges, on the Service Definition, page indicate whether to use a Windows service to run IBM Business Process Manager.

If the profile is configured as a Windows service, IBM Business Process Manager starts Windows services for processes started by the `startServer` or `startManager` commands. For example, if you configure a server as a Windows service and issue the `startServer` command, the `wasservice` command starts the defined services.

**Important:** If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, must belong to the Administrator group, and must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

#### **IPv6 (Internet Protocol version 6) considerations when running profiles as Windows services**

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the



service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

16. Advanced: Configure the Common database using a design file.
  - a. Select **Use a database design file**.
  - b. Click **Browse**.
  - c. Specify the fully qualified path name for the design file.
  - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.

**Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**. If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration.

Skip to the Business Space Configuration step

17. On the Database Configuration - Part 1 page, specify the information for the Common database.
  - a. From the **Select a database product** list, select **Microsoft SQL Server**.
  - b. Optional: If you want to change the default values, enter new names for the required databases. The database names must be unique.
  - c. To run the database scripts automatically as part of the profile creation process, select **Run database scripts to initialize the databases**. Otherwise, you or the database administrator must run the scripts manually after profile creation completes. You must also load the database with system information by running the **bootstrapProcessServerData** command.
  - d. Click **Next**.
18. On the Database Configuration - Part 2 page, complete the configuration for the Common database. Specify the following fields.

Table 215. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
Select the <b>Apply Windows Authentication</b> option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Process Server database, Performance Data Warehouse database, and Common database fields are made inactive.	
<b>Common database user name</b>	Enter the user name to authenticate with the database.
<b>Common database password</b>	Enter a password to authenticate with the database.
<b>Confirm password</b>	Confirm the password.
<b>Location (directory) of JDBC driver classpath files</b>	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: <code>\${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer</code>
<b>Database server host name (for example IP address)</b>	Accept the default value of localhost or enter the correct database server host name.
<b>Server port</b>	Accept the default value of 1433 or enter the correct server port number.

19. Advanced: On the Business Space Configuration page, if you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select **Configure IBM Forms Server** and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
20. Advanced: Select whether to configure a Business Process Rules Manager for the installation and click **Next**. Business Process Rules Manager is a web application that customizes the business rules templates for your business application needs.
21. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
22. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.
  - Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
  - Optionally, configure Business Process Choreographer.

*Augmenting deployment manager profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to augment existing WebSphere Application Server V8.0 custom profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root*\bin directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a deployment manager profile. You can determine the template by viewing the profile registry in the *install\_root*\properties\profileRegistry.xml file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.
 

Templates for each profile are located in the *install\_root*\profileTemplates\BPM directory for BPM templates and under *install\_root*\profileTemplates for other products. The following templates are available:

  - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard.
  - dmgr.procctr.adv: for a Process Center deployment manager profile for IBM BPM Advanced.
  - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard.
  - dmgr.procsvr.adv: for a Process Server deployment manager profile for IBM BPM Advanced.
  - dmgr.esbserver: for a WebSphere Enterprise Service Bus deployment manager profile.
3. Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions



are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the *install\_dir/profileTemplates/BPM* directory.

4. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

The samples are located in the *install\_root\BPM\samples\manageprofiles* directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response

- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2z0S.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2z0S.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:

```
manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/dmgr.procctr.adv
-profileName MyProfileName
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

- Add custom (managed nodes) profiles to be managed by the deployment manager and then configure the deployment environment.
- Optionally, configure Business Process Choreographer.

#### *Creating or augmenting custom profiles:*

As part of the network deployment configuration, you must create or augment at least one custom profile. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node.

#### *Creating custom profiles using the Profile Management Tool:*

You can create and federate custom profiles using the Profile Management Tool.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:****Vista****Windows 7**

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSPProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  4. On the **Profiles** tab, click **Create**.  
The Environment Selection page opens in a separate window.
  5. On the Environment Selection page, locate the IBM Business Process Manager Advanced configuration and expand the section. Select the profile to create and click **Next**.
  6. If you selected **Typical** profile creation, skip to the Federation step.
  7. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.  
When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one

profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
8. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

9. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

#### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

#### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

10. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
- To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
  - In the **Path** field, enter the directory path to the existing certificate.
  - In the **Password** field, enter the password for the certificate
  - In the **Keystore type** field, select the keystore type for the certificate you are importing.
  - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
  - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

11. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.securi ty file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

12. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

13. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
14. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
15. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

*Creating custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create custom profiles.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the *install\_root\bin* directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your profile (using the **-templatePath** option).  
Templates for each profile are located in the *install\_root\profileTemplates\BPM* directory for BPM templates and under *install\_root\profileTemplates* for other products. The following templates are available for custom profiles:
  - managed.procctr: for a Process Center custom profile for IBM BPM Standard.
  - managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
2. Determine which parameters are required for your profile by reviewing the example profile creation commands. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** parameters topic.
3. To use the **-responseFile** parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.



The samples are located in the *install\_root*\BPM\samples\manageprofiles directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response
- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2z0S.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2z0S.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2z0S.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2z0S.response



- PS\_Std\_Standalone\_Oracle.response
- PS\_Std\_Standalone\_SQLServer.response
- PS\_Adv\_Standalone\_DB2.response
- PS\_Adv\_Standalone\_DB2zOS.response
- PS\_Adv\_Standalone\_Oracle.response
- PS\_Adv\_Standalone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

4. Run the file from the command line. For example:

**Note:** The following example is an example only and shows optional parameters as well as required parameters. For a more complete set of parameters that can be changed, see the response file for creating custom profiles.

```
manageprofiles.bat -create -templatePath install_root/profileTemplates/BPM/managed.procctr.adv
-adminUserName bpmadmin -adminPassword bpmsecret -dbUserId dbuser -dbPassword dbsecret
```

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Response file for creating custom profiles:*

You can use the response file provided here with the **manageprofiles** command to create a custom profile.

*Sample response file for an Advanced Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Adv profile that uses SQLServer.
#
```

```

Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Adv_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#

```

```

Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true

```

```

Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a

```

```

message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#

```

```

Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None

```

```

#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the

```



```

-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer

```

```

#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Center custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PC Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PC_Std_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

```

```

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procctr

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is

```

```

the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name
#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword
#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies

```

```

Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None

```

```

#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic

```

```

disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or

```



```

stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production

```

```

Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for an Advanced Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Adv profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Adv_Managed_SQLServer.response
#

```

```

If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation
root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr.adv

#####
Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumber where profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileName where
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01

```

```

#####
Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

```

```

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:
An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true

```

```

Default Values:
None
#
Parameter: serviceName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The
localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winServiceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winServiceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this

```

```

parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic
disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbType
#
Description:
The database type. Set one of the following values for the type of
database product you are using with IBM Business Process Manager.
#
Required only for one of the following:
A stand-alone profile when -bpmdbDesign was not specified.
An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
MSSQLSERVER_MICROSOFT
Default Values:
MSSQLSERVER_MICROSOFT
#
Parameter: dbJDBCClasspath
#
Description:
The directory path location of JDBC driver files.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
Varies
Default Values:
None
#####
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

#####
Parameter: dbWinAuth

```



```

#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have
changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:

```

```

None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production
Stage
Default Values:
Test
#####
environmentType=Test
#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when

```

```

creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Sample response file for a Standard Process Server custom profile:*

Modify and use this response file to create a custom profile using the **manageprofiles** command.

**Note:** You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```

BEGIN COPYRIGHT

#
Licensed Materials - Property of IBM
5725-C94
(C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
This sample program is provided AS IS and may be used, executed, copied
and modified without royalty payment by customer (a) for its own
instruction and study, (b) in order to develop applications designed to
run with an IBM WebSphere product, either for customer's own internal
use or for redistribution by customer, as part of such an application,
in customer's own products.
#

END COPYRIGHT

#####
These response file has the applicable parameters for creating a
Managed PS Std profile that uses SQLServer.
#
Depending on your environment, you may need to change the default values.
#
To create a profile with this response file specify:
#
was.install.root/bin/manageprofiles -response PS_Std_Managed_SQLServer.response
#
If you use the -response parameter, it must be the only parameter on
the command-line. If you include any manageprofile parameters
(in addition to -response) on the command-line, the response file
is ignored. And default WebSphere Application server profile will be created .
#####

#####
Parameter: create
#
Description:
Creates the profile.
If you are creating a profile, this parameter is required.
#####
create

#####
Parameter: templatePath
#
Description:
Specifies the directory path to the template files in the installation

```

```

root directory. Within the profileTemplates directory are various
directories that correspond to different profile types and that vary with
the type of product installed. The profile directories are the paths that
you indicate while using the -templatePath option.
Use absolute paths. This parameter must exist as a directory and point to
a valid template directory.
#
When using the -templatePath parameter, specify the fully qualified file
path for the parameter.
#
Valid Values:
Varies
Default Values:
None
#####
templatePath=BPM/managed.procsvr
#####

Parameter: profileName
#
Description:
Specifies the name of the profile. Use a unique value when creating a
profile.
Each profile that shares the same set of product binaries must have a
unique name. The default profile name is based on the profile type and a
trailing number, for example:
profileType ProfileNumberwhere profileType is a value such as ProcSrv,
Dmgr, or Custom and ProfileNumber is a sequential number that creates a
unique profile name. The value for this parameter must not contain spaces
or characters that are not valid such as the following: *, ?, ", <, >, ,,
/, \, and |. The profile name that you choose must not be in use. On
Windows platforms: If the fully qualified path contains spaces, enclose
the value in quotation marks. The default value is based on the
install_root directory, the profiles subdirectory, and the name of the
file. For example, the default for profile creation is:
WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
file in the install_root/properties directory. The value for this
parameter must be a valid path for the target system and must not be
currently in use. You must have permissions to write to the directory.
#
Valid Values:
Varies
Default Values:
None
#####
profileName=Custom01
#####

Parameter: cellName
#
Description:
Specifies the cell name of the profile. Use a unique cell name for each
profile.
The default value for this parameter is based on a combination of the
short host name, the constant Cell, and a trailing number, for example:
if (DMgr)
shortHostNameCellCellNumber
else
shortHostNameNodeNodeNumberCell
where CellNumber is a sequential number starting at 01 and NodeNumber is
the node number that you used to define the node name.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies

```

```

Default Values:
Varies
#
Parameter: nodeName
#
Description:
Specifies the node name for the node that is created with the new profile.
Use a unique value within the cell or on the workstation. Each profile
that shares the same set of product binaries must have a unique node name.
This parameter is required for profile creation only with the
dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
The default value for this parameter is based on the short host name,
profile type, and a trailing number, for example:
if (DMgr)
shortHostNameCellManagerNodeNumber
else
shortHostNameNodeNodeNumber
where NodeNumber is a sequential number starting at 01.
The value for this parameter must not contain spaces or any characters
that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
Valid Values:
Varies
Default Values:
Varies
#####
cellName=cell_name
nodeName=node_name

#####
Parameter: enableAdminSecurity
#
Description:
For IBM Business Process Manager omit this parameter. Administrative
security is always enabled for IBM Business Process Manager profiles.
You must also specify the parameters -adminUserName and -adminPassword
along with the values for these parameters.
#
Valid Values:
true
Default Values:
true
#####
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

#####
Parameter: signingCertDN
#
Description:
Specifies the distinguished name of the root signing certificate that you
create when you create the profile. Specify the distinguished name in
quotation marks. This default personal certificate is located in the
server keystore file. If you do not specifically create or import a root
signing certificate, one is created by default. See the
-signingCertValidityPeriod parameter and the -keyStorePassword.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: signingCertValidityPeriod
#
Description:

```

```

An optional parameter that specifies the amount of time in years that the
root signing certificate is valid. If you do not specify this parameter
with the -signingCertDN parameter, the root signing certificate is valid
for 20 years.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: keyStorePassword
#
Description:
Specifies the password to use on all keystore files created during profile
creation. Keystore files are created for the default personal certificate
and the root signing certificate.
#
Valid Values:
Varies
Default Values:
None
#####
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
#####
Parameter: enableService
#
Description:
Enables the creation of a Linux service. Valid values include true or
false. The default value for this parameter is false. When the
manageprofiles command-line utility is run with the -enableService option
set to true , the Linux service is created with the profile when the
command is run by the root user. When a nonroot user runs the
manageprofiles command-line utility, the profile is created, but the Linux
service is not. The Linux service is not created because the nonroot user
does not have sufficient permission to set up the service. An
INSTCONPARTIALSUCCESS result is displayed at the end of the profile
creation and the profile creation log
install_root/logs/manageprofiles/profile_name_create.log contains a
message indicating the current user does not have sufficient permission to
set up the Linux service. Use this parameter when creating profiles only.
#
Valid Values:
false
true
Default Values:
None
#
Parameter: serviceUserName
#
Description:
Specifies the user ID that is used during the creation of the Linux
service so that the Linux service will run under this user ID. The Linux
service runs whenever the user ID is logged on. Not required.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceAccountType
#
Description:
The type of the owner account of the Windows service created for the
profile. Valid values include specifieduser or localsystem. The

```

```

localsystem value runs the Windows service under the local account of the
user who creates the profile. The default value for this parameter is
localsystem. Use this parameter when creating profiles only.
#
Valid Values:
localsystem
specifieduser
Default Values:
localsystem
#
Parameter: winserviceUserName
#
Description:
Specify your user ID so that the Windows operating system can verify you
as an ID that is capable of creating a Windows service. Your user ID must
belong to the Administrator group and have the following advanced user
rights:
Act as part of the operating system
Log on as a service
The default value for this parameter is the current user name. The value
for this parameter must not contain spaces or characters that are not
valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
you specify must have the proper permissions to create a Windows service.
You must specify the correct password for the user name that you choose.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winservicePassword
#
Description:
Specify the password for the specified user or the local account that is
to own the Windows service.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: winserviceCheck
#
Description:
The value can be either true or false. Specify true to create a Windows
service for the server process that is created within the profile. Specify
false to not create the Windows service. The default value for this
parameter is false.
#
Valid Values:
false
true
Default Values:
false
#
Parameter: winserviceStartupType
#
Description:
The following values for Windows service startup can be used:
manual
automatic
disabled
The default value for this parameter is manual.
#
Valid Values:
manual
automatic

```



```

disabled
Default Values:
manual
#####
#enableService=true
#serviceName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType>manual

#####
Parameter: dbWinAuth
#
Description:
An optional parameter that is valid in all default profile templates.
Specify true to indicate that you are running SQL Server using integrated
windows authentication.
#
Note: Do not use this parameter if you are using -bpmdbDesign.
#
Valid Values:
false
true
Default Values:
None
#####
#dbWinAuth=true

#####
Parameter: federateLaterBPM
#
Description:
Indicates if the managed profile is to be federated later using either the
profileTemplates/managed.esbserver or the
profileTemplates/BPM/managed.templateName template. Valid values are true
or false. If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
-dmgrAdminPassword parameters are not set, the default value for this
parameter is true. If you use -federateLaterBPM, you must either omit the
WebSphere Application Server-based parameter -federateLater or set it to
true.
#
Valid Values:
false
true
Default Values:
true
#
Parameter: dmgrHost
#
Description:
Identifies the workstation where the deployment manager is running.
Specify this parameter and the dmgrPort parameter to federate a custom
profile as it is created or augmented. This parameter is available with
the managed.templateName and managed.esbserver profile templates. The
host name can be the long or short DNS name or the IP address of the
deployment manager workstation. Specifying this optional parameter directs
the manageprofiles command-line utility to attempt to federate the custom
node into the deployment manager cell as it creates the custom profile.
This parameter is ignored when creating a deployment manager profile or
stand-alone server profile. If you federate a custom node when the
deployment manager is not running, the installation indicator in the logs
is INSTCONFFAILED to indicate a complete failure. The resulting custom
profile is unusable. You must move the custom profile directory out of the
profile repository (the profile's installation root directory) before
creating another custom profile with the same profile name. If you have

```

```

changed the default JMX connector type, you cannot federate with the
manageprofiles command-line utility. Use the addNode command later after
the node is created instead. The default value for this parameter is
localhost. The value for this parameter must be a properly formed host
name and must not contain spaces or characters that are not valid such as
the following: *, ?, ", <, >, , , /, \, and |. A connection to the
deployment manager must also be available in conjunction with the dmgrPort
parameter. Required if you are creating a managed profile with the
-federateLaterBPM parameter set to false and the deployment manager is not
on the local machine.
#
Valid Values:
Varies
Default Values:
None
#
Parameter: dmgrPort
#
Description:
Identifies the SOAP port of the deployment manager. Specify this parameter
and the dmgrHost parameter to federate a custom profile as it is created
or augmented. The deployment manager must be running and accessible. If
you have changed the default JMX connector type, you cannot federate with
the manageprofiles command-line utility. Use the addNode command later
after the node is created instead. The default value for this parameter is
8879. The port that you indicate must be a positive integer and a
connection to the deployment manager must be available in conjunction with
the dmgrHost parameter. Required if you are creating a managed profile,
the -federateLaterBPM parameter is set to false, and the deployment
manager was configured with a port that was not a default port.
#
Valid Values:
Varies
Default Values:
None
#####
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password
#####
Parameter: environmentType
#
Description:
Specifies the environment type. The environment type refers to how IBM
Business Process Manager is used; for example, in a production, stage or
test environment. Load testing might be done on a test server, while a
stage environment type might be used as a temporary location to host
changes before putting those changes into production. You might specify
Stage as the Environment type if the server you are configuring will be
accessed and used to review content and new functionality. Valid values
are as follows:
Test
Use Test if the server you are configuring is to be used as a testing
environment.
Stage
Use Stage if the server is to serve as a staging platform to be used as a
preproduction server.
Production
Use Production if the server is to serve in a production capacity.
The default value is Test.
#
Valid Values:
Test
Production

```

```

Stage
Default Values:
Test
#####
environmentType=Test

#####
Parameter: isDeveloperServer
#
Description:
Specifies whether the server is intended for development purposes only.
This parameter is useful when creating profiles to test applications on a
non-production server prior to deploying the applications on their
production application servers. If -isDeveloperServer is set when
creating an IBM Business Process Manager profile, then a preconfigured VMM
file repository is installed. This file repository contains a sample
organization that can be used to test Business Process Choreographer
people resolution, ready for you to use as is.
#
Valid Values:
false
true
Default Values:
false
#####
#isDeveloperServer=false

```

*Augmenting custom profiles using the Profile Management Tool:*

If you have existing WebSphere Application Server V8.0 custom profiles, you can augment an existing profile using the Profile Management Tool to add support for IBM Business Process Manager.

Remember to shut down any servers associated with a profile you plan to augment.

The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Enter the following command:

```
install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar
```

For example, to start the Profile Management Tool in the German language, enter the following command:

```
install_root\java\bin\java -Duser.language=de install_root\bin\ProfileManagement\startup.jar
```

**Restrictions:** Vista Windows 7

- To run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or Access is denied message occurs in the `install_root\bin\ProfileManagement\pmt.bat` file. By default, nonadministrative users do not have

access to the Program Files directory, which is the default installation location for the product. To resolve this issue, nonadministrative users must either install the product by themselves or be given permission to access the other product instances.

After you start the Profile Management Tool, you must decide whether to choose **Typical** or **Advanced** profile creation. Use the advanced option to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
  - Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.
1. If you want to federate the custom node to a deployment manager while creating the custom profile, start the deployment manager.
  2. Use one of the following methods to start the Profile Management Tool.
    - Start the tool from the First steps console.
    - Use the Windows Start menu. For example, select **Start > Programs or All Programs > IBM > Business Process Manager 8.0 > Profile Management Tool**.
    - Run the command `install_root\bin\ProfileManagement\pmt.bat`
  3. Shut down any servers associated with a profile you plan to augment.
  4. On the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
  5. On the **Profiles** tab, select the profile that you want to augment and click **Augment**. If you augment a WebSphere Application Server profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented. The Augment Selection page opens in a separate window.
  6. On the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.
  7. On the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

**Restriction:** The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
  - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
  - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
8. If you selected **Typical** profile creation, skip to the Federation step.
  9. Advanced: On the Profile Name and Location page, perform the following steps:
    - a. In the **Profile name** field, specify a unique name or accept the default value. Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name.
    - b. In the **Profile directory** field, enter the directory for the profile or use the **Browse** button to go to the profile directory. The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is `install_root\profiles\profile_name`.
    - c. Optional: Select **Make this profile the default** to make the profile you are creating the default profile. This check box is shown only if you have an existing profile on your system.

When a profile is the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies.

- d. From the **Server runtime performance tuning setting** list, select a performance tuning level appropriate for the profile you are creating. This parameter is a WebSphere Application Server parameter.
  - e. Click **Next**. If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.
10. Advanced: On the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
- In the **Node name** field, enter a name for the node or accept the default value. Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment.
  - In the **Server name** field, enter a name for the server or accept the default value.
  - In the **Host name** field, enter a name for the host or accept the default value.
  - In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next**.

11. On the Federation page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

#### **Important:**

Select **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

#### **Processing associated with federating the node as part of custom profile creation:**

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** and then make different selections on the Federation page.

Click **Next**. If you selected **Typical** profile creation, skip to the Database Configuration step.

12. Advanced: On the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.
- To create a new default personal certificate and a new root signing certificate, select **Create a new default personal certificate** and **Create a new root signing certificate**, and click **Next**.
  - To import existing certificates, select **Import an existing default personal certificate** and **Import an existing root signing personal certificate** and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. Advanced: On the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify. An imported certificate is added to the key.p12 file or the root-key.p12 file. If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. Advanced: On the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**. The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.



Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

```
profile_root\properties\portdef.prop
```

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see "Updating ports in existing profiles" in the WebSphere Application Server information center. Run the **updatePorts.ant** file through the **ws\_ant** script detailed in this topic.

15. On the Database Configuration page, select the database used by the deployment manager and confirm the location of the JDBC driver classpath files.
16. On the Profile Summary page, review the information. Click **Create** to create the profile or **Back** to change the characteristics of the profile.
17. On the Profile Complete page, review the information. To proceed to the First Steps Console, make sure that **Launch First Steps Console** is selected and click **Finish**.

After you have finished adding custom profiles, configure the deployment environment.

#### Related information:

 [Updating ports in existing profiles \(WebSphere Application Server\)](#)

*Augmenting custom profiles using the manageprofiles command-line utility:*

Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to augment existing WebSphere Application Server V8.0 profiles.

Remember to shut down any servers associated with a profile you plan to augment.

Make sure that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed when you run the command, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

If the profile that you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.

**Restriction:** To run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the `install_root\bin` directory:

```
runas /env /user:MyAdminName "manageprofiles.bat -response myResponseFile"
```

Non-administrative users are prompted for the administrator password.

1. Determine the template that was used to create the existing profile that you want to augment. You must augment a custom profile. You can determine the template by viewing the profile registry in the `install_root\properties\profileRegistry.xml` file. Do not modify this file; use it only to view the templates.
2. Find the appropriate template to use for the augmentation.  
Templates for each profile are located in the `install_root\profileTemplates\BPM` directory for BPM templates and under `install_root\profileTemplates` for other products. The following templates are available for custom profiles:
  - `managed.procctr`: for a Process Center custom profile for IBM BPM Standard.



- managed.procctr.adv: for a Process Center custom profile for IBM BPM Advanced.
  - managed.procsvr: for a Process Server custom profile for IBM BPM Standard.
  - managed.procsvr.adv: for a Process Server custom profile for IBM BPM Advanced.
  - managed.esbserver: for a WebSphere Enterprise Service Bus custom profile.
3. Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the `manageprofiles` command-line utility to update or augment the profile identified in the `-profileName` parameter using the template in the `-templatePath` parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for `-templatePath`, because a relative file path for the `-templatePath` parameter results in the specified profile not being fully augmented.

**Note:** Do not manually modify the files that are located in the `install_dir/profileTemplates/BPM` directory.

4. To use the `-responseFile` parameter, look at the provided sample response files and create a response file that is specific to your environment.

**Important:** Ensure that you do not leave blank spaces after the values; for example `'personalCertValidityPeriod=1 '` or `'winserviceCheck=false '`. Spaces will cause profile creation to fail. The samples are located in the `install_root\BPM\samples\manageprofiles` directory. Choose an appropriate response file based on the database and type of profile (stand-alone or network, Process Center or Process Server). The following sample files are available:

- PC\_Std\_DMGr\_DB2.response
- PC\_Std\_DMGr\_DB2z0S.response
- PC\_Std\_DMGr\_Oracle.response
- PC\_Std\_DMGr\_SQLServer.response
- PC\_Adv\_DMGr\_DB2.response
- PC\_Adv\_DMGr\_DB2z0S.response
- PC\_Adv\_DMGr\_Oracle.response
- PC\_Adv\_DMGr\_SQLServer.response
- PC\_Std\_Managed\_DB2.response
- PC\_Std\_Managed\_DB2z0S.response
- PC\_Std\_Managed\_Oracle.response
- PC\_Std\_Managed\_SQLServer.response
- PC\_Adv\_Managed\_DB2.response
- PC\_Adv\_Managed\_DB2z0S.response
- PC\_Adv\_Managed\_Oracle.response
- PC\_Adv\_Managed\_SQLServer.response
- PC\_Std\_StandAlone\_DB2.response
- PC\_Std\_StandAlone\_DB2z0S.response
- PC\_Std\_StandAlone\_Oracle.response
- PC\_Std\_StandAlone\_SQLServer.response
- PC\_Adv\_StandAlone\_DB2.response
- PC\_Adv\_StandAlone\_DB2z0S.response
- PC\_Adv\_StandAlone\_Oracle.response
- PC\_Adv\_StandAlone\_SQLServer.response
- PS\_Std\_DMGr\_DB2.response
- PS\_Std\_DMGr\_DB2z0S.response

- PS\_Std\_DMGr\_Oracle.response
- PS\_Std\_DMGr\_SQLServer.response
- PS\_Adv\_DMGr\_DB2.response
- PS\_Adv\_DMGr\_DB2zOS.response
- PS\_Adv\_DMGr\_Oracle.response
- PS\_Adv\_DMGr\_SQLServer.response
- PS\_Std\_Managed\_DB2.response
- PS\_Std\_Managed\_DB2zOS.response
- PS\_Std\_Managed\_Oracle.response
- PS\_Std\_Managed\_SQLServer.response
- PS\_Adv\_Managed\_DB2.response
- PS\_Adv\_Managed\_DB2zOS.response
- PS\_Adv\_Managed\_Oracle.response
- PS\_Adv\_Managed\_SQLServer.response
- PS\_Std\_StandAlone\_DB2.response
- PS\_Std\_StandAlone\_DB2zOS.response
- PS\_Std\_StandAlone\_Oracle.response
- PS\_Std\_StandAlone\_SQLServer.response
- PS\_Adv\_StandAlone\_DB2.response
- PS\_Adv\_StandAlone\_DB2zOS.response
- PS\_Adv\_StandAlone\_Oracle.response
- PS\_Adv\_StandAlone\_SQLServer.response

Copy one of the sample response files to your working directory. Edit the parameters in the response file as appropriate to your configuration and save the edited response file. Make sure that the **templatePath** property path matches your specific installation directory.

5. Run the file from the command line. Do not supply a **-profilePath** parameter. For example:  
`manageprofiles.bat -augment -templatePath install_root/profileTemplates/BPM/managed.procctr.adv -profileName MyProfileName`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles.sh -response myResponseFile
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

After you have finished adding custom profiles, configure the deployment environment.

*Federating custom nodes to the deployment manager:*

After creating a custom node, you can use the **addNode** command to federate the custom node into a deployment manager cell. You can manage all federated nodes from the deployment manager.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created a deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.

- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile\_root* represents the installation location of the deployment manager profile:
 

```
profile_root\bin\startManager.bat
```
  - The deployment manager has been created or augmented to be an IBM Business Process Manager deployment manager.
  - The deployment manager is at the same release level or higher than the custom profile you created or augmented.
  - The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
  - You do not plan to use this custom node as a migration target.
1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to the following directory (from a command line), where *profile\_root* represents the installation location of the custom profile):
 

```
profile_root\bin
```

2. Run the **addNode** command.

Run the following command from the command line if security is not enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port
```

Run the following command from the command line if security is enabled:

```
addNode.bat deployment_manager_host deployment_manager_SOAP_port -username
userID_for_authentication -password password_for_authentication
```


An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

```
ADMU0003I: Node DMNDID2Node03 has been successfully federated.
```

The custom profile is federated into the deployment manager.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

#### Related information:

 Add managed node settings (WebSphere Application Server)

#### Loading the database with system information in a network deployment environment:

If you are creating a network deployment environment, you must run the command before you try to start or use Process Server or Process Center.

**Important:** Before you run the bootstrap command, ensure that you have applied Mandatory Interim Fix JR44669. For more information, see Required interim fix for APAR JR44669. To download the fix, see Required interim fixes for IBM Business Process Manager.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the Process Server database. This data is required for the BPM applications to run correctly

- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. For a cluster, you need to specify the cluster name. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.
- If a single WebSphere cell contains multiple application target clusters, you must run this command on each of the clusters.

Run the bootstrap utility from the command line. The bootstrap utility is found in the deployment manager profile directory. For example:

```
Linux UNIX install_root/profiles/Dmgr01/bin
Windows install_root\profiles\Dmgr01\bin
```

Run the bootstrap utility using one of the following commands:

- **bootstrapProcessServerData.sh -clusterName *cluster\_name***
- **bootstrapProcessServerData.sh -nodeName *node\_name* -serverName *server\_name***

where:

- **-clusterName** is the name of the application target cluster. You must specify this parameter when you want the bootstrap data to run on a cluster.
- **-nodeName** is the name of the node. You must specify this parameter and the **-serverName** parameter when you want the bootstrap data to run on the server that is part of the network deployment environment and not part of the cluster.
- **-serverName** is the name of the server. You must specify either this parameter and the **-nodeName** parameter when you want the bootstrap data to run on the server that is part of the network deployment and not part of the cluster.

The parameters are case-sensitive.

**Important: For SQL Server only:** Ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL\_Latin1\_General\_CP1\_CI\_AS** (not **SQL\_Latin1\_General\_CP1\_CS\_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
```

```
Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement
(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute
(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery
(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

You have loaded the database with system information before successfully starting the IBM Business Process Manager server. The log information for the bootstrap operation is saved to the `USER_INSTALL_ROOT/logs/` directory in a file called `bootstrapProcessServerData.clusterName.timestamp.log` or `bootstrapProcessServerData.nodeName.serverName.timestamp.log` depending on the target you specified. The console displays a subset of the logged information.

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData
```

Bootstrap data onto a cluster that hosts the Process Server or Process Center:

```
bootstrapProcessServerData -clusterName myAppCluster
```

**Modifying an existing installation:**

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

